



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Industrial

Desarrollo e implementación de herramientas heurísticas
para la programación del calendario de exámenes de la
ETSII considerando la distancia entre aulas

Trabajo Fin de Grado

Grado en Ingeniería de Organización Industrial

AUTOR/A: Del Río Ortuño, Ramón

Tutor/a: Esteso Álvarez, Ana

Cotutor/a: Alemany Díaz, María del Mar

CURSO ACADÉMICO: 2023/2024

AGRADECIMIENTOS

Me gustaría agradecer a mis padres por permitirme poder estudiar esta carrera y a mi familia por el significativo apoyo que me han transmitido a lo largo de todos mis años de estudio y lo orgulloso que me han hecho sentir, en especial mención a mi abuelo Joaquín, ejemplo de perseverancia y dedicación, al cual le hubiese encantado verme graduado como Ingeniero.

A su vez, hago mención especial a mi tutora Ana Estesos, sin la cual no podría haber realizado esta propuesta. Su gran dedicación, ayuda y entusiasmo, han sido los pilares fundamentales que han hecho todo posible.

RESUMEN

Este Trabajo Fin de Grado (TFG) se centra en el desarrollo e implementación de herramientas heurísticas para la programación del calendario de exámenes ordinarios de la Escuela Técnica Superior de Ingeniería Industrial (ETSII) de la Universitat Politècnica de València (UPV). En la ETSII se imparten 5 titulaciones de Grado, 1 titulación de Doble Grado y 9 titulaciones de Máster. Algunas de estas titulaciones cuentan además con un elevado número de estudiantes matriculados. La capacidad limitada de las aulas hace necesaria la utilización simultánea de múltiples aulas en un mismo examen para poder albergar a todos sus estudiantes matriculados.

La ETSII de la UPV cuenta además con aulas en varios edificios y varios pisos. Esto hace que sea problemático para los profesores de una asignatura gestionar el examen cuando las aulas asignadas a su examen se encuentran en diferentes pisos e incluso edificios. Por ello, este TFG se centra en desarrollar heurísticas para la programación de exámenes y la asignación de aulas, considerando la distancia e incluso la conexión de las aulas, tratando de minimizar la distancia existente entre las aulas asignadas a un mismo examen.

Adicionalmente, en la ETSII se imparten algunas asignaturas en diversas titulaciones que requieren que su examen tenga lugar en el mismo día y hora por compartir examen. En este TFG, a estas asignaturas, se le llaman asignaturas equivalentes. De esta manera, las herramientas desarrolladas tienen en consideración que las asignaturas equivalentes deben ser programadas en el mismo día y turno, y en caso de ser posible, se agrupan sus estudiantes para poder emplear el menor número de aulas posibles.

Las herramientas heurísticas desarrolladas se implementan en el lenguaje de programación Visual Basic en Excel. Estas se validan a través de su aplicación al caso real de la ETSII de la UPV.

Palabras Clave: Heurística; Programación de exámenes; Asignación de aulas; Asignaturas equivalentes; Distancia entre aulas; ETSII

RESUM

Este Treball fi de grau (TFG) se centra en el desenvolupament i implementació de ferramentes heurístiques per a la programació del calendari d'exàmens ordinaris de l'Escola Tècnica Superior d'Enginyeria Industrial (ETSII) de la Universitat Politècnica de València (UPV). En la ETSII s'impartixen 5 titulacions de Grau, 1 titulació de Doble Grau i 9 titulacions de Màster. Algunes d'estes titulacions compten a més amb un elevat nombre d'estudiants matriculats. La capacitat limitada de les aules fa necessària la utilització simultània de múltiples aules en un mateix examen per a poder albergar a tots els seus estudiants matriculats.

La ETSII de la UPV compta a més amb aules en diversos edificis i diversos pisos. Això fa que siga problemàtic per als professors d'una assignatura gestionar l'examen quan les aules assignades al seu examen es troben en diferents pisos i fins i tot edificis. Per això, este *TFG se centra en desenvolupar heurístiques per a la programació d'exàmens i l'assignació d'aules, considerant la distància i fins i tot la connexió de les aules, tractant de minimitzar la distància existent entre les aules assignades a un mateix examen.

Addicionalment, en la ETSII s'impartixen algunes assignatures en diverses titulacions que requerixen que el seu examen tinga lloc en el mateix dia i hora per compartir examen. En este *TFG, a estes assignatures, se li diuen assignatures equivalents. D'esta manera, les ferramentes desenvolupades tenen en consideració que les assignatures equivalents han de ser programades en el mateix dia i torn, i en cas de ser possible, s'agrupen els seus estudiants per a poder emprar el menor nombre d'aules possibles.

Les ferramentes heurístiques desenvolupades s'implementen en el llenguatge de programació Visual Basic en Excel. Estes es validen a través de la seua aplicació al cas real de la ETSII de la UPV.

Paraules Clau: Heurística; Programació d'exàmens; Assignació d'aules; Assignatures equivalents; Distància entre aules; ETSII.

ABSTRACT

This Final Degree Project focuses on the development and implementation of heuristic tools for the scheduling of the exam calendar of the Escuela Técnica Superior de Ingeniería Industrial (ETSII) of the Universitat Politècnica de València (UPV). The ETSII offers 5 Bachelor's Degrees, 1 Double Degree and 9 Master's Degrees. Some of these degrees also have a high number of students enrolled. The limited capacity of the classrooms makes it necessary to use multiple classrooms at the same time for the same exam in order to be able to accommodate all the students enrolled.

The ETSII of the UPV also has classrooms in several buildings and on several floors. This makes it problematic for the teachers of a subject to manage the exam when the classrooms assigned to their exam are located on different floors and even buildings. Therefore, this TFG focuses on developing heuristics for exam scheduling and classroom assignment, considering the distance and even the connection of the classrooms, trying to minimize the distance between the classrooms assigned to the same exam.

In addition, the ETSII teaches some subjects in different degrees that require the exam to take place on the same day and at the same time due to the sharing of exams. In this TFG, these subjects are called equivalent subjects. In this way, the tools developed take into consideration that equivalent subjects should be scheduled on the same day and shift, and if possible, students are grouped together in order to use as few classrooms as possible.

The heuristic tools developed are implemented in the Visual Basic programming language in Excel. These are validated through their application to the real case of the ETSII of the UPV.

Keywords: Heuristics; Examination scheduling; Classroom allocation; Equivalent subjects; Distance between classrooms; ETSII

CONTENIDO

1. Memoria
2. Presupuesto



MEMORIA

DESARROLLO E IMPLEMENTACIÓN DE HERRAMIENTAS HEURÍSTICAS PARA
LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES DE LA ETSII
CONSIDERANDO LA DISTANCIA ENTRE AULAS

TRABAJO FIN DE GRADO EN GRADO EN INGENIERÍA DE ORGANIZACIÓN
INDUSTRIAL

AUTOR: del Río Ortuño, Ramón

TUTOR: Esteso Álvarez, Ana

COTUTORA: Díaz Alemany, Mareva

CURSO ACADÉMICO: 2023/24

ÍNDICE DE LA MEMORIA

Contenido

Índice de figuras	10
Índice de tablas	12
1. OBJETO, MOTIVACIÓN Y JUSTIFICACIÓN	13
1.1. Objeto.....	13
1.2. Motivación.....	14
1.3. Justificación	15
1.3.1. Justificación profesional.....	15
1.3.2. Justificación académica.....	16
1.4. Estructura del TFG.....	16
1.4. Competencias Transversales UPV	17
2. DESCRIPCIÓN DE LA PROBLEMÁTICA	20
2.1. Contexto: la Escuela Técnica Superior de Ingeniería Industrial	20
2.2. Descripción de la problemática.....	21
2.3. Descripción del problema en el periodo de exámenes.....	23
2.4. Propuesta de solución.....	24
2.5. Impacto en la sostenibilidad	25
3. ANTECEDENTES	27
3.1. Introducción	27
3.2. Exam Timetable Problem (ETP)	27
3.3. Revisión de heurísticas para la programación del calendario de exámenes	30
3.3.1. Contextualización.....	31
3.3.2. Heurísticas Avanzadas aplicadas al ETP	32
3.3.2.1. Heurística directa	33
3.3.2.2. Algoritmos genéticos.....	33
3.3.2.3. Búsqueda tabú	34
3.3.2.4. Heurísticas constructivas.....	35
3.3.2.5. Algoritmos de búsqueda local.....	38
4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS.....	40
4.1. Introducción	40
4.2. Hipótesis de partida	40

4.2. Definición de la problemática	41
4.3. Modelo de Heurística Constructiva para la programación de exámenes considerando la distancia entre aulas.....	42
4.3.1. Base de datos y Restricciones	42
4.3.2. Criterio o Regla de selección	46
4.3.3. Algoritmo paso a paso.....	47
4.3.3.1. Inicialización	47
4.3.3.2. Construcción de la solución.....	49
4.3.3.3. Finalización.....	53
4.4. Arquitectura de la herramienta heurística constructiva para la programación de exámenes ordinarios.....	54
4.5. Conclusiones.....	55
5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS	57
5.1. Introducción	57
Hipótesis.....	57
5.2. Definición de la problemática	57
5.3. Modelo de Algoritmo de Búsqueda Local para la programación de exámenes que minimiza la suma de las distancias de las aulas asignadas a la asignatura objeto	57
5.3.1. Base de datos y Restricciones	57
5.3.2. Algoritmo de Búsqueda Local.....	58
5.3.2.1. Solución inicial.....	58
5.3.2.2. Generación de vecinos	58
5.3.2.3. Función de evaluación y criterio de mejora	60
5.3.2.4. Condición de parada	60
5.3.2.5. Algoritmo paso a paso.....	60
5.4. Arquitectura de la herramienta heurística de búsqueda local para la programación de exámenes	64
5.6. Conclusiones.....	65
6. APLICACIÓN A LA ETSII. ANÁLISIS DE RESULTADOS	67
6.1. Introducción	67
6.2. Solución de la Heurística Constructiva	67
6.3. Solución del Algoritmo de Búsqueda local	71
Asignatura 11472 Física I.....	71
Asignatura 12280 Proyectos	71
Asignatura 12939 Ciencia de Materiales.....	71
6.4. Comparación de resultados.....	74

6.5. Conclusiones.....	77
7. FUTURAS LÍNEAS DE ACTUACIÓN Y CONCLUSIONES	79
7.1. Futuras líneas de actuación.....	79
7.2. Conclusiones finales	80
BIBLIOGRAFÍA.....	83
ANEXO	85
Anexo I. Código de la heurística constructiva	86
Anexo II. Código de la función Calcular_Tamaño.....	101
Anexo III. Código de la función Leer_Tabla_Col y Leer_Disponibilidad	102
Anexo IV. Código de la función Ordenar_Matriz.....	105
Anexo V. Código de la heurística de búsqueda local.....	106
Anexo VI. Código de la función GenerarAleatorio	143
Anexo VII. Horario Solución.....	144

Índice de figuras

Figura 1. Plano 2D de edificios de la ETSII (Fuente: adaptado de https://www.upv.es/plano/plano-2d-es.html).....	21
Figura 2. Esquema factores que influyen en la programación del periodo de exámenes (Fuente: Elaboración propia).....	22
Figura 3. Esquema organización exámenes por titulación (Fuente: Elaboración propia).....	24
Figura 4. Esquema de los tipos de ETP. (Fuente: Elaboración propia).....	29
Figura 5. Esquema de los tipos de ETP. Fuente: (Ceschia, Di Gaspero, & Schaerf, 2022).....	30
Figura 6. Esquema de los pasos para la modelización de una problemática (Fuente: Elaboración propia).....	31
Figura 7. Resumen de la base de datos del caso de la UMP. Fuente: (Kahar & Kendall, 2014)..	36
Figura 8. Ejemplo de tabla de datos de las asignaturas (Fuente: Elaboración propia)	43
Figura 9. Ejemplo de tabla de datos de las asignaturas Equivalentes (Fuente: Elaboración propia)	43
Figura 10. Ejemplo de tabla de datos de las aulas (Fuente: Elaboración propia)	45
Figura 11. Ejemplo de tabla de datos de la distancia entre las aulas (Fuente: Elaboración propia)	45
Figura 12. Diagrama de flujo de Inicialización (Fuente: Elaboración propia).....	48
Figura 13. Diagrama de flujo de Construcción de la solución (Fuente: Elaboración propia)	51
Figura 13. Diagrama de flujo de Construcción de la solución (Fuente: Elaboración propia)	52
Figura 14. Diagrama de flujo de Finalización (Fuente: Elaboración propia)	54
Figura 15. Arquitectura Técnica de la herramienta heurística para la programación de exámenes (Fuente: Elaboración propia).....	54
Figura 16. Diagrama de Flujo del Algoritmo Paso a Paso (Fuente: Elaboración propia).....	62
Figura 17. Diagrama de Flujo del Algoritmo Paso a Paso (Fuente: Elaboración propia).....	63
Figura 18. Arquitectura Técnica de los Algoritmos de búsqueda local para la programación de exámenes (Fuente: Elaboración propia)	64
Figura 19. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia).....	69
Figura 20. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia).....	70
Figura 21. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)	72

Figura 22. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)..... 73

Figura 23. Gráfica comparativa de la eficiencia de las heurísticas (Fuente: Elaboración propia) 74

Tabla 1. Comparación de eficiencia de las heurística (Fuente: Elaboración propia..... 76

Anexo VII. Figura 1. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia) 145

Anexo VII. Figura 2. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)..... 146

Anexo VII. Figura 3. Horario Solución Heurística Constructiva para el Cuatrimestre B (turno de mañana) (Fuente: Elaboración propia) 147

Anexo VII. Figura 4. Horario Solución Heurística Constructiva para el Cuatrimestre B (turno de tarde) (Fuente: Elaboración propia)..... 148

Anexo VII. Figura 5. Horario Solución Algoritmo de Búsqueda Total con 5 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 149

Anexo VII. Figura 6. Horario Solución Algoritmo de Búsqueda Total con 5 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 150

Anexo VII. Figura 7. Horario Solución Algoritmo de Búsqueda Total con 10 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 151

Anexo VII. Figura 8. Horario Solución Algoritmo de Búsqueda Total con 10 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 152

Anexo VII. Figura 9. Horario Solución Algoritmo de Búsqueda Total con 50 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 153

Anexo VII. Figura 10. Horario Solución Algoritmo de Búsqueda Total con 50 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 154

Anexo VII. Figura 11. Horario Solución Algoritmo de Búsqueda Total con 100 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 155

Anexo VII. Figura 12. Horario Solución Algoritmo de Búsqueda Total con 100 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 156

Anexo VII. Figura 13. Horario Solución Algoritmo de Búsqueda Total con 250 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 157

Anexo VII. Figura 14. Horario Solución Algoritmo de Búsqueda Total con 250 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 158

Anexo VII. Figura 15. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 159

Anexo VII. Figura 16. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 160

Anexo VII. Figura 17. Horario Solución Algoritmo de Búsqueda Total con 1000 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 161

Anexo VII. Figura 18. Horario Solución Algoritmo de Búsqueda Total con 1000 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 162

Anexo VII. Figura 19. Horario Solución Algoritmo de Búsqueda Total con 5000 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)..... 163

Anexo VII. Figura 20. Horario Solución Algoritmo de Búsqueda Total con 5000 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia) 164

Índice de tablas

Tabla 1. Comparación de eficiencia de las heurísticas. Fuente: Elaboración propia 76

1. OBJETO, MOTIVACIÓN Y JUSTIFICACIÓN

1.1. Objeto

El fundamento general de este Trabajo Fin de Grado (TFG) consiste en el diseño e implementación de herramientas de investigación operativa para la programación de exámenes ordinarios universitarios, tomando como referencia la casuística particular de la Escuela Técnica Superior de Ingeniería Industrial (ETSII) de la Universitat Politècnica de València (UPV).

El propósito de las herramientas desarrolladas es proporcionar una solución factible a la asignación de fechas y aulas para la realización de los exámenes con el objetivo de reducir la distancia entre las aulas asignadas a un mismo examen. Esto implica, por un lado, asegurar la correcta asignación de las materias impartidas teniendo en cuenta el curso al que pertenecen, si cuentan con asignaturas cuyo examen se deba programar para el mismo día y turno, el número de alumnado matriculado y el profesorado disponible; por otro lado, aplicar modelos heurísticos que permitan gestionar eficientemente los recursos y minimizar los conflictos en la programación propuesta; y, en adición novedosa, tomar en consideración la distancia entre las aulas y las restricciones logísticas y académicas que afecten a la institución.

Entre las principales funcionalidades de la herramienta a desarrollar se incluyen:

- **Presentar una asignación factible de fechas y horarios.** Se busca conseguir el calendario que mejor se adapte a las necesidades de cada grado, curso, cuatrimestre y materia, teniendo en cuenta tanto al alumnado como al profesorado.
- **Logística de las aulas.** Atender a las peticiones del profesorado relacionadas con la proximidad de las aulas, pudiendo así tener mejor control de la realización de los exámenes y del alumnado y que el alumnado posea mayor comodidad.
- **Facilitar la gestión y la planificación.** Proveer una interfaz intuitiva y amigable para el profesorado encargado de la programación de exámenes, permitiendo la introducción de datos, la configuración de parámetros y la visualización de resultados de manera eficiente.

La herramienta de investigación operativa propuesta consta de dos heurísticas para la programación de exámenes ordinarios. La primera es una heurística constructiva que pretende proporcionar una programación de exámenes ordinarios factible en la que se considera la compatibilidad de exámenes de un mismo curso, la compatibilidad de exámenes de cursos sucesivos, la necesidad de programar ciertos exámenes simultáneamente, la distancia existente entre las aulas y el ajuste de las aulas empleadas a las necesidades de cada asignatura. En segundo lugar, una heurística de búsqueda local que, partiendo de la solución inicial proporcionada por la heurística constructiva, pretende mejorar esta primera programación buscando reducir la suma de las distancias existente entre las aulas asignadas a cada asignatura. Ambas heurísticas obtienen soluciones de una manera eficiente y ofrece una solución intuitiva, fácil de interpretar y de aplicar a la realidad de la ETSII.

Para que la programación sea satisfactoria se han de cumplir ciertas condiciones fundamentales como son el no solapar asignaturas del mismo curso o cursos contiguos (por si hay alumnos que han de repetir materias) dentro del mismo grado, el poder concentrar asignaturas cuyo examen sea igual (conocidas como asignaturas equivalentes) o el factor diferencial de esta propuesta que es que las aulas asignadas a un mismo examen puedan disponer de la mayor proximidad posible para facilitar tanto a alumnos como docentes el correcto curso de los exámenes.

Tanto la heurística constructiva como la heurística de búsqueda local se han implementado en la herramienta de Microsoft Office dentro de su módulo de programación Visual Basic, una herramienta muy sólida que proporcionará la solidez a nuestra solución. Además, la herramienta de MS Excel es ampliamente conocida por los usuarios docentes, por lo que permitirá una interfaz cómoda y de fácil utilización.

Por último, cabe destacar que durante la validación de la herramienta propuesta se han utilizado datos reales de la ETSII de la UPV, para la puesta en valor en un caso real de la herramienta de investigación operativa.

1.2. Motivación

La elección del tema objeto de este TFG nace de la necesidad de una herramienta que pueda ayudar tanto a los docentes como a los alumnos. A su vez, la complejidad que conlleva la asignación del gran número de asignaturas y de alumnos de los diferentes grados a un número limitado de días en el cuatrimestre A o B del curso académico hace que se cree un nicho de mejora. Debido a la citada complejidad, se ha decidido implementar también dos algoritmos que pretenda mejorar la solución propuesta.

Partiendo de este punto, otro factor que es considerado un “rompecabezas” para los docentes debido a la escasez, en algunos casos, de suficientes profesores o espacios disponibles, ha sido el implementar la problemática de las clases próximas o hasta conectadas.

Recientemente se han llevado a cabo estudios en relación a la problemática de la asignación de los horarios de los exámenes, esta línea de investigación se denomina Exam Timetabling Problems (ETP). En la literatura existen diversas herramientas que tratan de abordar este problema y, en muchas de las que lo abordan mediante modelos de optimización es difícil encontrar una solución óptima en un tiempo razonable debido su complejidad. Por ello, se abre la oportunidad de la utilización de heurísticas para poder hallar solución a este complejo problema en un tiempo razonable. Así, con este TFG se pretende poder avanzar en este campo y poder proporcionar una solución fácilmente aplicable a los centros docentes universitarios.

Si atendemos a nuestro caso práctico concreto, la ETSII de la UPV, imparte diferentes grados y postgrados de la misma índole. Debido al gran número de asignaturas repartidos en los diferentes títulos, la limitación de recursos docentes y de aulas disponibles, y la recurrencia del proceso de exámenes, nace la necesidad de una respuesta objetiva que pueda satisfacer los reclamos de profesores y alumnos que permitan mejorar las metas propuestas ayudando, a su vez, a los tres pilares de la sostenibilidad:

- Dimensión social. Con la correcta organización de los periodos de exámenes se pretende mejorar la disponibilidad tanto del alumnado, permitiéndole así desempeñar mejor las pruebas; y a los docentes, permitiéndoles una mayor facilidad de acceso a las diferentes pruebas de una forma sencilla.
- Dimensión económica. Mediante la programación propuesta, el coste es una mínima inversión en recursos tanto humanos como materiales. Además, debido a la reducción del consumo de electricidad y de energía relacionado con la disminución de las aulas utilizadas, se produce un ahorro en el coste energético.
- Dimensión medioambiental. Al favorecer con esta investigación al mínimo uso de aulas y priorizar las aulas, de forma indirecta se está favoreciendo al ahorro energético de las salas, lo que supone una contribución no solo económica sino también, medioambiental.

1.3. Justificación

Tras haber expresado el objeto y la motivación de la realización de este Trabajo de Fin de Grado, en último lugar de este capítulo, se pretende justificar la realización del mismo. Este apartado se ha dividido en dos partes, una primera que hace referencia a la justificación profesional, y una segunda parte que hace referencia a la justificación académica.

1.3.1. Justificación profesional

La justificación profesional proviene de poder dar una solución aplicable, fácil e intuitiva, a un problema cotidiano que resulta bastante complejo para todas las Escuelas Técnicas Superiores de Ingeniería y Facultades de las Universidades del país y que, además, está relacionado con los conocimientos que se pretenden de un Ingeniero de Organización Industrial.

Aunque cada Universidad, Facultad o Escuela se rige por sus propias normativas, existe una legislación común relacionada con nuestro campo de investigación que son las pruebas de exámenes. Estas normas individualizadas son las que particularizan el caso para la ETSII en concreto, pero que también dificultan y hacen de una forma más compleja la resolución de la problemática.

En definitiva, la razón profesional del TFG radica en optimizar mediante el uso de técnicas relacionadas con las de un profesional de la Ingeniería de Organización Industrial la disposición de los títulos de grado y de postgrado de una Escuela o Facultad en relación a las pruebas examinadoras. Mediante esto se pretende una mejor gestión tanto de profesores como de estudiantes del periodo de exámenes.

1.3.2. Justificación académica

En primer lugar, la razón principal consiste en que la realización del TFG es el último requisito para la obtención del grado, destacando también la necesidad de analizar desde la perspectiva de la Ingeniería de Organización Industrial, un problema cotidiano como es la programación de los exámenes universitarios. Para ello, se han aplicado diferentes conocimientos y competencias adquiridos a lo largo de la realización del grado, complementándolos aún más con la profundización en la línea de la investigación operativa relacionada con las heurísticas.

Por otra parte, durante la realización del TFG se he tenido que aplicar los conocimientos y habilidades adquiridos en diversas asignaturas cursadas durante el Grado en Ingeniería de Organización Industrial:

- Matemáticas I. Como bloque básico de cualquier ingeniería permite adquirir conocimientos matemáticos que facilitan posteriormente la comprensión de diferentes asignaturas, permitiendo poder analizar y desarrollar soluciones.
- Matemáticas II. Forma parte, a su vez, del bloque de asignaturas base de la ingeniería y facilita la comprensión de asignaturas futuras, permitiendo poder analizar y desarrollar soluciones.
- Informática. Sirve como base de programación para la realización del código en el lenguaje de Visual Basic y la correcta utilización de MS Excel.
- Planificación y Control de Producción y Operaciones. Permite tener una idea sobre los conceptos de programación y del uso de heurísticas en diferentes ámbitos.
- Métodos Cuantitativos para la Organización Industrial. Es la base de este TFG, ya que en esta asignatura se enseña todo lo relativo a las heurísticas y su amplio abanico de utilidades.

1.4. Estructura del TFG

La memoria del TFG se estructura en siete capítulos. Como se ha podido ver, el capítulo “1. OBJETO, MOTIVACIÓN Y JUSTIFICACIÓN” incluye el objeto, motivación y justificación del TFG, así como un análisis de aquellas Competencias Transversales sobre las que se ha trabajado durante el desarrollo del TFG.

El capítulo “2. DESCRIPCIÓN DE LA PROBLEMÁTICA”, aborda la descripción y contextualización de la ETSII en la que se enmarca el TFG, la descripción de la problemática de la programación de exámenes, y la descripción de la propuesta de solución que en este TFG se trata. Así, en este segundo capítulo se puede contextualizar al lector sobre el porqué de abordar dicho problema y la solución propuesta.

Continuando con la puesta en situación al lector, en el capítulo “3. ANTECEDENTES” se trata la parte teórica referida tanto a las herramientas de investigación operativa utilizadas, es decir, las heurísticas, como las investigaciones realizadas en el campo de la programación de exámenes universitarios. Por último, se trata la explicación teórica de las heurísticas constructiva y de búsqueda local que dan sentido a la propuesta.

En el capítulo “4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS”, se desarrolla en sí misma la primera propuesta realizada en este TFG de una heurística constructiva que permita la asignación de las pruebas a los diferentes huecos de horarios y se alinee con la necesidad de que las clases se encuentren lo más próximas posibles.

Complementario a este apartado, se encuentra el capítulo “5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS”, que, teniendo como base la solución encontrada con la heurística constructiva, la mejora utilizando los Algoritmos de Búsqueda Local que permiten la obtención de soluciones vecinas a partir de esta.

Para completar la propuesta de las herramientas heurísticas, estas se implementan en Visual Basic, y se validan con el caso real de la ETSII de la UPV, encarnada en el capítulo “6. APLICACIÓN A LA ETSII. ANÁLISIS DE RESULTADOS”.

Tras haber presentado las herramientas heurísticas de solución propuestas, se presenta el capítulo “7. FUTURAS LÍNEAS DE ACTUACIÓN Y CONCLUSIONES FINALES”, en la que se realiza una síntesis sobre la efectividad de las herramientas propuestas y las futuras líneas de investigación a raíz de estas, proponiéndose líneas de mejora.

Por último, en el capítulo “BIBLIOGRAFÍA” se presenta la bibliografía que refleja las fuentes de información utilizadas para la realización de este TFG.

1.4. Competencias Transversales UPV

Una vez descrito cómo se ha estructurado el documento, se enumeran las competencias transversales que se han aplicado, aprendido y desarrollado a lo largo del TFG. Para ello, nos centramos en las cinco Competencias transversales establecidas por la UPV (Universitat Politècnica de València, 2024):

CT_01. Responsabilidad y toma de decisiones. Con la realización de un trabajo de este tipo se requiere de una conciencia sobre la responsabilidad ya que, el TFG es resultado de un trabajo constante en el que se toman diferentes decisiones que desembocan en el trabajo final presentado. Los resultados del aprendizaje de la misma son:

- ✓ “Identificar, formular y resolver problemas complejos, de manera autónoma, aplicando los principios de la disciplina, como es la programación de exámenes”.
- ✓ “Desarrollar y realizar trabajos e investigaciones, prácticas o experimentales, interpretando datos y extrayendo conclusiones fundamentadas en los principios de la disciplina”, siendo esto el desarrollo y las labores de investigación que conllevan la realización del TFG.
- ✓ “Adquirir y aplicar nuevos conocimientos según sea necesario, utilizando estrategias de aprendizaje y de gestión del tiempo apropiadas”, ya que para poder realizar el TFG se han tenido que adquirir conocimientos relacionado con la programación en Visual Basic de MS Excel y ampliar los conocimientos sobre los Métodos Cuantitativos.
- ✓ “Aplicar de manera efectiva técnicas relacionadas con la búsqueda bibliográfica y el uso de fuentes de datos fiables u otros sistemas de información”.

CT_02. Comunicación efectiva. Con la defensa y la realización del documento se pretende que llegue a diferentes públicos facilitando la comprensión de un problema con gran complejidad como este. Los resultados del aprendizaje de la misma son:

- ✓ “Estructurar el discurso para favorecer la comprensión de los objetivos, acciones y/o resultados de un trabajo propio”, dividiendo el documento en capítulos que facilitan la lectura del documento.
- ✓ “Desarrollar textos profesionales o informes científico-técnicos según las convenciones propias de la disciplina”, como este documento.
- ✓ “Comunicar y argumentar eficazmente, adaptando la organización de contenidos y el uso del lenguaje, verbal y no verbal, a diversas situaciones y/o ante diversas audiencias”, pudiendo defender ante el tribunal la idea expuesta en este TFG.
- ✓ “Demostrar destreza en la comunicación digital utilizando medios de apoyo variados y adaptados a la situación y a la audiencia”, siendo ejemplos de esto el documento Word y PDF que representan el informe y el material de apoyo para la defensa.

CT_03. Trabajo en equipo y liderazgo. Al realizar de forma complementaria con la tutora de este trabajo el mismo, se ha favorecido al trabajar en equipo, focalizándose en el objetivo e intentando mejorar. Los resultados del aprendizaje de la misma son:

- ✓ “Funcionar eficazmente en un equipo cuyos miembros juntos brinden liderazgo y creen un entorno colaborativo e inclusivo en la organización y coordinación del trabajo”.
- ✓ “Identificar los roles y destrezas para operar en equipos multidisciplinares con diferentes perfiles profesionales”.
- ✓ “Colaborar de manera proactiva en el desarrollo del trabajo, estableciendo metas y cumpliendo objetivos”.
- ✓ “Contribuir a la búsqueda de soluciones a retos o proyectos, demostrando empatía y asertividad a la hora de compartir ideas, reflexiones y argumentos en el seno del trabajo colaborativo”.

CT_04. Innovación y creatividad. Mediante el desarrollo de una herramienta funcional, intuitiva y aplicable se ha desarrollado una idea original que antes no se había profundizado. Los resultados del aprendizaje de la misma son:

- ✓ “Identificar nuevos retos, proyectos u oportunidades de mejora en el ámbito de la disciplina alineados con tendencias y avances futuros”, actividad base la hora de plantear una necesidad que pueda ser empleada como objeto de un TFG.
- ✓ “Proponer soluciones creativas para responder satisfactoriamente a necesidades y problemas reales de la sociedad”, siendo reflejo de esto las diferentes heurísticas que se presentan a lo largo del documento y que pretenden responder a la problemática de la programación de exámenes.
- ✓ “Evaluar, de manera crítica y constructiva, las ventajas y las oportunidades de diferentes soluciones a un mismo problema”, comparando la eficiencia de las diferentes heurísticas propuestas viendo que solución se ajusta más a las necesidades del problema.
- ✓ “Demostrar una actitud emprendedora en el diseño, desarrollo y ejecución de soluciones que supongan una novedad o avance en el ámbito de la disciplina”, al intentar progresar en una línea de actuación como es la aplicación de heurísticas al problema de la programación de exámenes.

CT_05. Compromiso social y medioambiental. Como se ha comentado anteriormente, al intentar reducir la utilización de las aulas se reduce el consumo energético y en consecuencia un ahorro de daño medioambiental. Los resultados del aprendizaje de la misma son:

- ✓ “Valorar las consecuencias éticas de las decisiones a tomar en una situación concreta, considerando el impacto en la sociedad y la responsabilidad en la práctica profesional”
- ✓ “Emitir juicios informados considerando el impacto de las soluciones, en el ámbito de la disciplina, en contextos globales, económicos, sociales y medioambientales”.
- ✓ “Demostrar concienciación sobre el respeto a la diversidad y a los principios de accesibilidad universal y diseño para todas las personas”.
- ✓ “Diseñar, desarrollar y ejecutar soluciones en el ámbito de la disciplina, que den respuesta a demandas sociales, teniendo en cuenta como referente los Objetivos de Desarrollo Sostenible y factores globales, culturales, y económicos”, explicando la relación de la actividad propuesta en este TFG con los ODS.

2. DESCRIPCIÓN DE LA PROBLEMÁTICA

2.1. Contexto: la Escuela Técnica Superior de Ingeniería Industrial

La Escuela Técnica Superior de Ingeniería Industrial (ETSII) forma parte del conjunto de catorce escuelas y facultades que conforman la Universidad Politécnica de Valencia (UPV). La historia de la creación de esta institución se haya en la creación de la UPV en 1968, cuando se instituyó el Instituto Politécnico Superior de Valencia (IPSV) debido al Decreto ley 5/1968 de 6 de junio de 1968 de Medidas Urgentes de Reestructuración Universitaria y cuatro meses después, en el Decreto 2731/1968 de 24 de octubre de 1968, se determinan los centros que constituirán el IPSV, encontrándose entre ellos la ETSI (UPV, 2024).

La ETSII lleva más de 50 años formando a ingenieros de diferentes ramas del ámbito industrial, ayudando a la Comunidad Valenciana a saciar la necesidad de profesionales de este ámbito. La institución es perteneciente al gremio de las Universidades públicas españolas y ofrece la posibilidad de los siguientes grado y cursos de postgrado (ETSII-UPV, 2024):

- Grado en Ingeniería en Tecnologías Industriales (GITI).
- Grado en Ingeniería Química (GIQ).
- Grado en Ingeniería de la Energía (GIE).
- Grado en Ingeniería Biomédica (GIB).
- Grado en Ingeniería de Organización Industrial (GIOI).
- Doble Grado en Ciencia de Datos + Ingeniería de Organización Industrial (CD+GIOI).
- Máster Universitario en Ingeniería Industrial (MUII).
- Máster Universitario en Ingeniería Química (MUIQ).
- Máster Universitario en Tecnología Energética para el Desarrollo Sostenible (MUTEDS).
- Máster Universitario en Ingeniería Biomédica (MUIB).
- Máster Universitario en Ingeniería Avanzada de Producción, Logística y Cadena de Suministro (MUIAPLCS).
- Máster Universitario en Dirección y Gestión de Proyectos (MUDGP).
- Máster Universitario en Construcciones e Instalaciones Industriales (MUCII).
- Máster Universitario en Seguridad Nuclear y Protección Radiológica (MUSNPR).
- Máster Universitario en Seguridad Industrial (MUSI).

Al igual que se ha citado anteriormente la diferenciación de créditos que encarnan cada una de las diferentes titulaciones, también cada una de ellas ofrece un número diferentes de plazas. Mientras que el grado de GITI ofrece 275 plazas, siendo el grado más numeroso, los grados de GIQ y GIOI ofrecen 80 plazas, y, por último, los grado en GIE y GIB son dotados de 75 plazas (ETSII-UPV, 2024). El recientemente añadido doble grado de CD+GIOI cuenta solamente con 30 plazas.

En el caso de los cursos de postgrado, el número de matriculados es bastante inferior, proporcionando 60 plazas para el MUIQ y MUIB, siendo 45 el dotado para el MUTEDS, 30 plazas para el MUIAPLC y el MUSI, y de 25 plazas para los MUCII y MUDGP y de 20 para el MUSNPR. Solo sobresale el MUII que es el más grande de la escuela teniendo un número esperado de matrículas al año que oscila los 300 alumnos (ETSII-UPV, 2024).

Lo que estos datos sostienen es que la ETSII recoge cada año alrededor de 2600 alumnos de grado, sin tener en cuenta alumnos que sostengan matrículas de segunda y tercera matrícula, y unos 750 alumnos de curso de postgrado, lo que supone más de 4100 alumnos anuales entre ambos tipos de docencia (ETSII-UPV, 2024). Además, cabe añadir la cifra de los profesores que sobrepasa las siete centenas y que provienen de al menos 26 Departamentos.

Aunque la UPV tiene un campus en Vera (Valencia), otro en Gandía y en Alcoi, nuestro objeto de estudio estará centrado en el campus de Vera, dado que en él se encuentra la ETSII. La escuela está dotada de diferentes edificios a lo largo de la ciudad universitaria, teniendo su sede de secretario en el 5H y el edificio más emblemático reside en el aulario del 5N, por ser el más espacioso y con el mayor número de aulas. La escuela cuenta con 14 edificios que representan alrededor de 46.000 metros cuadrado, coloreados en la Figura 1 con color fucsia. Los matriculados y docentes disponen de 42 aulas de teoría, 12 aulas informáticas y 6 salas para presentaciones con diferentes capacidades (ETSII-UPV, 2024)

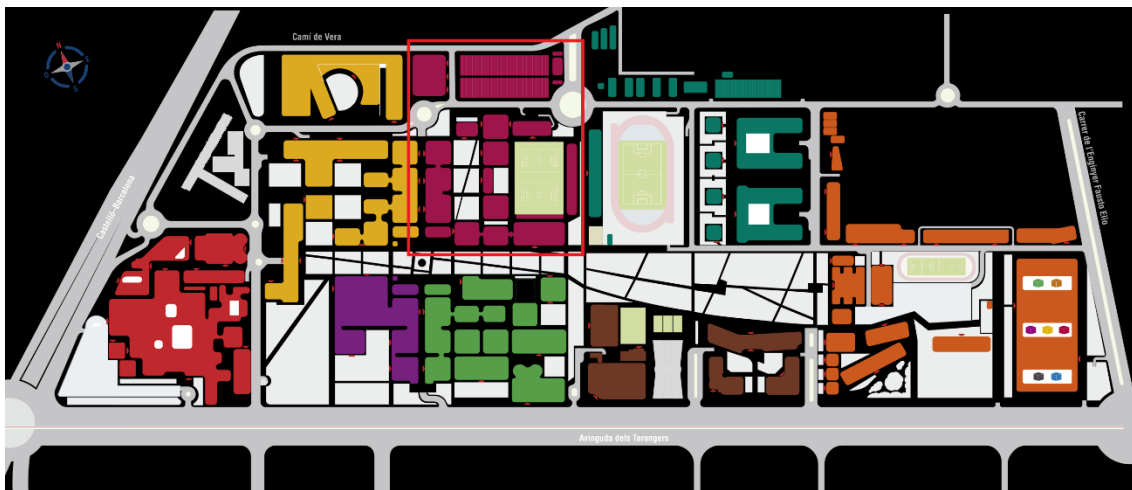


Figura 1. Plano 2D de edificios de la ETSII (Fuente: adaptado de <https://www.upv.es/plano/plano-2d-es.html>)

2.2. Descripción de la problemática

El fundamento de la realización de esta investigación es dar solución al problema de la programación de exámenes de una Escuela o Facultad en el ámbito universitario, centrándonos como ejemplo de caso real en la ETSII de la UPV en sendos periodos de exámenes de ambos cuatrimestres en convocatoria ordinaria, estableciendo un periodo predefinido.

A forma de resumen, se ilustra en la Figura 2 los principales factores que influyen en la programación de exámenes.

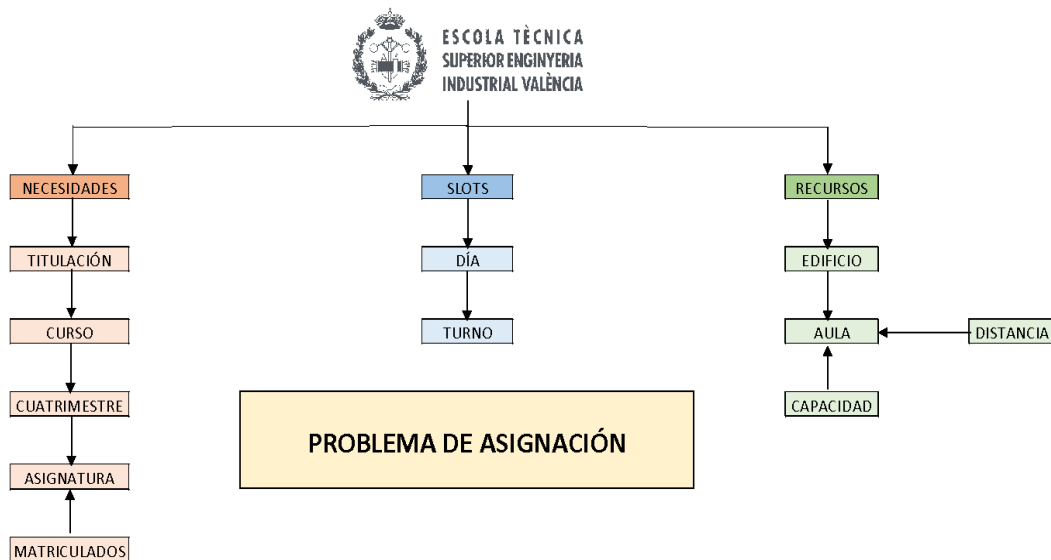


Figura 2. Esquema factores que influyen en la programación del periodo de exámenes (Fuente: Elaboración propia)

Como se puede apreciar en la Figura 2, existen factores de gran relevancia que influyen en la programación de las pruebas: las necesidades, que hacen referencia a las asignaturas y los datos que respectan de esta; los “slots” que es lo que se conoce como el espacio de tiempo que se asigna a los exámenes; y los recursos, es decir, las aulas con sus cualidades que la definen.

En primer lugar, las necesidades se definen como lo que respecta al alumnado y a las asignaturas. Se definen las asignaturas, objeto principal de la asignación de los exámenes y que tienen como características fundamentales, la titulación a la que pertenecen, el curso al que hacen referencia, el cuatrimestre y el número de matriculados. Esta característica es la consideramos como clave, ya que es la que definirá las aulas donde se realizarán los exámenes.

Como segundo bloque se encontrarían los “slots” o espacios de tiempo en el que se asignará una asignatura, que se realizará en un/as aula/s en concreto. Este factor es clave ya que, dependiendo del cuatrimestre en el que se encuentre una asignatura, se asignará en un turno de mañana o de tarde y, dependiendo de la disponibilidad, de un día u otro

Por último, se encuentran las aulas, este factor es el más complejo de la problemática ya que hay que tener en cuenta que el aula se puede asignar hasta dos veces por día; aunque tiene un aforo predeterminado, para la realización de los exámenes este se reduce a la mitad; y, como punto diferenciador a la hora de resolver la problemática se ha tenido en cuenta el priorizar las clases que se encuentren más cercanas, pudiendo ayudar a los docentes a tener un mayor control a la hora de la realización de las pruebas y proporcionando a los alumnos una mayor comodidad.

2.3. Descripción del problema en el periodo de exámenes

La problemática de la planificación y programación de los exámenes se trata de un difícil proceso en el que se asigna el examen de una asignatura a un horario o “slot” determinado en un día hora, un aula o aulas donde se realizarán las pruebas, todo ello para cada titulación y curso durante los dos periodos semestrales que definen el calendario docente.

Esta problemática es conocida por ser tan compleja debido a que hay que tener en cuenta múltiples factores. Como se comentaba en la sección anterior y ya que los recursos tanto docentes como de instalaciones en cualquier institución como la ETSII son limitados surge un dilema a la hora de asignar las asignaturas. Desarrollando esta idea, cada asignatura posee un número determinado de profesores que imparten la misma y que son responsables de la misma y, por tanto, cuantas menos clases se asignen menor número de profesores se necesitarán y se podrán cubrir los exámenes con profesores especializados en la asignatura en cuestión. Por otro lado, las aulas poseen una capacidad también limitada por lo que no todas pueden albergar un examen y, en algunos casos se ha de hacer uso de más de una por lo que eso hay que tenerlo en cuenta a la hora de tener que resolver el problema. Si se desarrolla la idea de priorizar la proximidad de las aulas, se minimiza el movimiento tanto de profesores como de alumnos entre las aulas y, como consecuencia se minimiza también la posibilidad de plagio.

En relación a las asignaturas, la existencia de asignaturas equivalentes hace que se complique la asignación ya que se ha de tener en cuenta que puede estar ya asignada siendo su equivalente de otra titulación y, en algunos casos, hasta de diferentes cursos. Para tener en cuenta esto se ha de sumar dichos matriculados y asignarlos en el mismo intervalo de tiempo, priorizando la asignatura con mayor número de matriculados.

Cabe añadir que, aunque se ha citado anteriormente, puede darse el caso de que parte del alumnado se encuentre cursando asignaturas de diferentes cursos dentro de la misma titulación por lo que hay que priorizar que no se produzca el solape entre las asignaturas de diferentes cursos. Para esto se ha decidido asignar las asignaturas de los cursos 1º y 3º en un turno, mientras que los cursos de 2º y 4º realizarán sus exámenes en el turno contrario.

Atendiendo al proceso que utiliza la ETSII actualmente, este consiste en reutilizar el calendario de exámenes del año anterior, pero realizando una rotación de un día en adelante, es decir, si el segundo día del calendario de GIOI del segundo curso del primer cuatrimestre se realizaba el examen de la asignatura de Termodinámica, este año se realizará el tercer día, y así con todas las demás asignaturas de los diferentes cursos de los grados y másteres ofertados en la Escuela. En el caso de las asignaturas de idiomas, en este estudio las obviaremos, puesto que, en su mayoría son impuestas por la Centro de Lenguas de la UPV no por la ETSII.

Una vez definida la secuencia, se realiza de forma manual la asignación de las aulas, lo cual es bastante ineficiente si tenemos en cuenta la dificultad que supone esta problemática.

Por último, cabe destacar que actualmente solo cambia esta secuencia si se da de baja una asignatura o se añade una nueva. La exportación/importación de un curso a otro se realiza mediante la aplicación “ALGAR HORARIOS”.

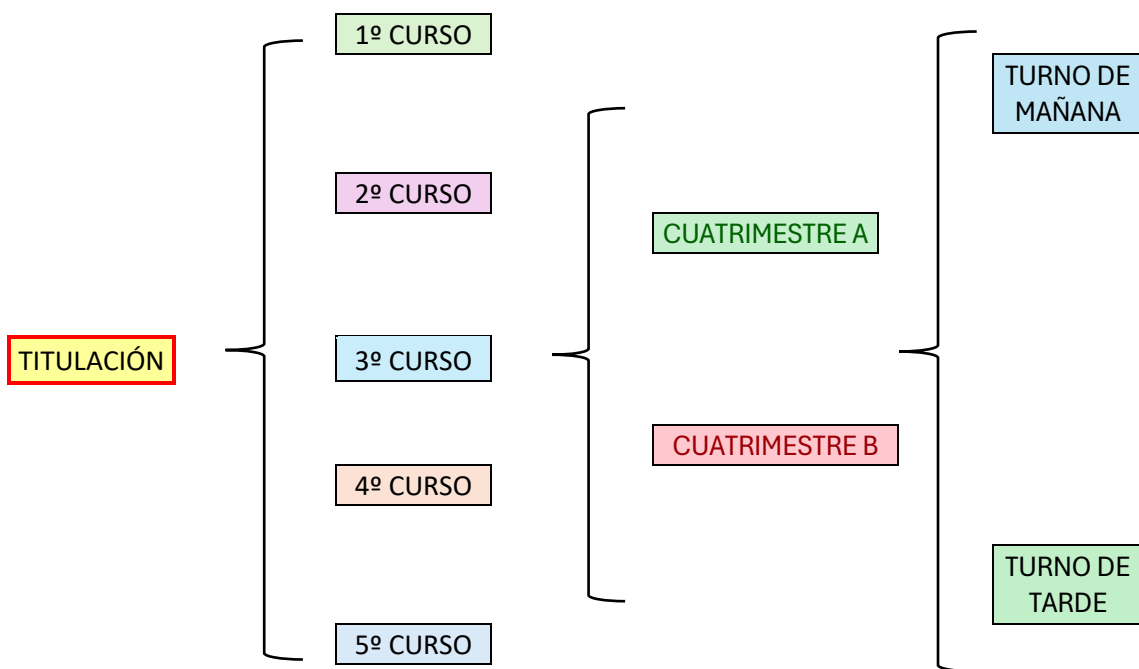


Figura 3. Esquema organización exámenes por titulación (Fuente: Elaboración propia)

Debido a que el proceso de programación de exámenes ordinarios actual puede mejorarse, se destacan las oportunidades de mejora:

1. Considerar la distancia existente entre las aulas asignadas a un mismo examen.
2. Ajustar la capacidad asignada a cada examen a la capacidad requerida según el número de matriculados.
3. Proporcionar medidas de eficiencia que permitan calificar la programación propuesta.
4. Facilitar una interfaz intuitiva y de fácil para la programación de exámenes.

2.4. Propuesta de solución

Una vez aportado las bases de la problemática y los puntos a mejorar, se procede a explicar la propuesta de solución a lo expuesto. Para solucionar o mejorar el actual método de programación de los exámenes se propone la utilización de una heurística constructiva en una primera instancia para la obtención de una solución inicial que tenga en cuenta la proximidad de las aulas y, a continuación, la permutación de la misma mediante un algoritmo de localización que baraje diferentes opciones basándose en la solución inicial antes obtenida.

Comenzando con la heurística citada, antes de comenzar la asignación, hay que agrupar y ordenar todas las asignaturas de la ETSII de las diferentes titulaciones de mayor a menor número de matriculados. A su vez, se ordenan las 42 clases disponibles en todas las instalaciones de la Escuela también de mayor a menor número de capacidad de asistentes de examen.

Una vez realizado este primer paso, se comienza la asignación. Se escoge en primer lugar la asignatura con mayor número de matriculados y se comprueba si, con la suma de las capacidades de las clases que hay disponible se puede asignar ese día y, si es así, se comprueba si tiene asignaturas equivalentes y se dan dos casos:

1. No tiene asignaturas equivalentes y, por tanto, se puede comenzar la asignación.
2. Tiene asignaturas equivalentes y, en este caso, se suma la cantidad de matriculados del total y se toma como si fuera una única asignatura, registrando a su vez el listado de asignaturas sobre el que se realiza la asignación.

Así, en cualquiera de los dos casos se comienza la asignación de la asignatura a la clase con mayor capacidad. Partiendo de esto se intenta asignar en las clases más próximas a la consecuente. Siendo esto de forma resumida el funcionamiento de la heurística, aunque se desarrollará más detalladamente en el Capítulo 5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS.

En esta fase inicial se tiene en cuenta la asignación de las aulas y de los recursos disponibles que en ese momento se encuentran en la Escuela y las aulas disponibles para la realización de las pruebas, utilizando así el menor número de recursos y produciendo un ahorro energético para la universidad y una mejor organización para el profesorado.

A continuación, entra en acción los Algoritmos de Búsqueda Local, que permitirán la generación de nuevas soluciones a raíz de la solución inicial y que pretenden minimizar la distancia entre las aulas asignadas a las asignaturas teniendo como función objetivo la suma y la media de estas distancias.

En definitiva, la solución propuesta tiene en cuenta todos los factores expuestos al principio del capítulo, considerando la minimización de las aulas y profesorado, pudiendo permitir la existencia de exámenes realizados en clases compartidas o prácticamente contiguas desde todo lo posible, permitiendo a los alumnos que no quepa la posibilidad de solape entre sus asignaturas y mejorando así el duro periodo examinador.

2.5. Impacto en la sostenibilidad

En este apartado, como conclusión se pretende desarrollar el impacto sostenible para la Escuela de la propuesta realizada.

En relación a la sostenibilidad medioambiental, mediante la aplicación de la heurística, se va a permitir el mayor aprovechamiento de las aulas de los diferentes edificios por lo que esto permitirá un menor consumo energético y de luz, ahorrando y ayudando a la menos contaminación de nuestro medio natural. Además, esto repercutirá en un ahorro económico importante que supondrá la mayor disposición de liquidez económica para la universidad que podría desembocar en la utilización de esta en la mejora de las instalaciones.

En cuanto al alumnado y al profesorado, estas medidas permitirán que, en el caso de los docentes mejore su organización y disponibilidad para el mejor aprovechamiento de su jornada laboral ya que la mayoría se encuentran en el ámbito de la investigación y les permitirá, a su vez, tener un mayor control en la doctrina anti-plagio y permitiendo el mejor curso de la realización de los exámenes. En el caso de los alumnos, en adición a la reducción de casos de solape de asignaturas, permitirá que se reduzca la presión producida debido al poco tiempo entre exámenes que se traducirá en un mejor rendimiento por parte de estos.

Tras haber expuesto el contexto de la situación actual, en el siguiente capítulo se va a abordar los antecedentes teóricos y científicos de la investigación operativa tanto en el ámbito de las heurísticas como en el de Exam Timetabling Problems (ETP).

3. ANTECEDENTES

3.1. Introducción

En este capítulo se va a profundizar en los estudios teóricos en los que se apoya este TFG, tratando los estudios sobre las herramientas utilizadas con el objetivo de revisar los avances en esta área de la investigación operativa y poner así en contexto al lector.

En primer lugar, se abordará la explicación de la problemática de este TFG, el ETP, contextualizando su historia, sus principales características y factores que influyen en él. A continuación, se realizará una contextualización sobre las heurísticas haciendo énfasis, sobre las heurísticas constructivas y de búsqueda local y sus investigaciones referenciándose a artículos, libros y revistas científicas que profundizan en los temas de interés. Además, se hablará, sobre los modelos heurísticos empleados en este ámbito.

Esta breve puesta en contacto con los asuntos más teóricos permite la justificación del TFG y el uso de las herramientas de investigación operativas utilizadas.

3.2. Exam Timetable Problem (ETP)

De acuerdo con (Astaiza A., 2005), la historia de la aplicación de los métodos computacionales a los problemas de programación de horarios se remonta a mediados del siglo XX. La primera generación se lanzó en los inicios de la década de los sesenta con el objetivo de reducir la cantidad de trabajo administrativo relacionado. Los enfoques heurísticos en la programación fueron introducidos por Broker y Cole en 1964. En 1967, Welsh y Powell señalaron una similitud entre este problema y el de colorear un grafo. Muchos algoritmos han utilizado la coloración de grafos como parte esencial del sistema desde las observaciones de Welsh y Powell en 1967.

HOREX fue un programa de computadora desarrollado por Desroches, Laporte y Rousseau en 1978 para la programación de exámenes, al principio, el programa encontraba una coloración del gráfico de conflictos y luego intentaba igualar la cantidad de exámenes en cada periodo. Para reducir el número de exámenes consecutivos que se asignan a las aulas, los periodos son ordenados y los fines de semana se incorporan porque normalmente no hay una secuencia para los exámenes. Los algoritmos para la asignación de aulas y la coloración de grafos se presentaron por Burke, Elliman y Were en 1993. Para tratar de solucionar los problemas intratables, integraron el modelo con un sistema de hoja electrónica para que el usuario pueda obtener una guía informada. El proceso de encontrar un periodo para cada examen, tal que no haya conflictos, es equivalente a asignar colores a vértices de un grafo de manera que los vértices adyacentes tengan colores diferentes (Astaiza A., 2005).

Muchos algoritmos han sido propuestos, la mayoría de los cuales emplean heurísticas basadas en la coloración de grafos. Estos pueden producir resultados adecuados, pero suelen ignorar que si hay suficientes asientos o no para todos los exámenes asignados en un periodo específico y no permiten la realización de ninguna investigación del espacio de soluciones. Por ello, se ha ido ampliado la investigación sobre el tema a lo largo de los últimos años, destacando las investigaciones sobre la aplicación de las Metaheurísticas como la Búsqueda Tabú, los Algoritmos Genéticos y los Algoritmos de Búsqueda Local (Astaiza A., 2005).

La programación se define como la asignación de recursos a lo largo del tiempo para realizar un conjunto de tareas y cuyo objetivo es asignar un conjunto de entidades a un número limitado de recursos a lo largo del tiempo, de forma que se cumpla un conjunto de requisitos de programación predefinidos. En los últimos años, se observa un patrón notable en el área de la programación académica ya sean escuelas o universidades, y puede dividirse en dos tipos: la programación del horario de las clases y la programación del horario de los exámenes. En un primer caso, es la programación de los cursos universitarios y de los profesores por semanas con el fin de eliminar los conflictos en los horarios y, por el contrario, la programación de exámenes consiste en asignar en un número limitado de franjas horarias disponibles las pruebas, cumpliendo con una serie de restricciones (Gashgari, y otros, 2018).

El problema de la programación académica se considera tanto un problema no determinista de tiempo polinómico duro (NP-duro) como un problema no determinista de tiempo polinómico completo (NP-completo), lo que significa que el tiempo de cálculo aumenta exponencialmente a medida que crece el tamaño del problema (Wibowo, Teoh, & Ngadiman, 2013). En la programación de exámenes, los exámenes se asignan a franjas horarias en un periodo determinado, la mayoría de los problemas de las programaciones están relacionados con las instituciones educativas y vienen dados por coordinar los cuatro recursos principales, que son los estudiantes y los profesores, las aulas y, por último, las franjas horarias para programar los cursos. Una vez que se termina la programación de los exámenes, el horario de exámenes mostrará la información sobre el examen/curso, la hora del examen, el lugar del examen y, en ocasiones, los supervisores o profesores (Siew, y otros, 2024).

El éxito de una solución de programación de horarios se mide por el grado en que satisface una serie de restricciones preestablecidas, las restricciones relativas al problema de la programación académica se pueden clasificar en dos categorías: restricciones duras y restricciones blandas. Básicamente, las restricciones duras son restricciones obligatorias que no pueden ser violadas bajo ninguna circunstancia en absoluto, de lo contrario, el horario no sea factible. Según la (Abdipoor, Yaakob, Goh, & Abdullah, 2023). Por otro lado, las restricciones blandas, como las preferencias de los profesores, son restricciones secundarias que pueden violarse y, aunque son restricciones secundarias, se prefiere que no se violen, ya que para la eficacia y la calidad de la solución son importantes. Las restricciones duras y blandas suelen variar en función de la institución, algunas de las restricciones duras más comunes en los problemas de programación son: que ningún alumno puede ser asignado a dos o más exámenes en el mismo periodo, que la disponibilidad de recursos suficientes es limitada para todos los exámenes o la asignación de un número adecuado de profesores en función del número de estudiantes en la sala (Abdipoor, Yaakob, Goh, & Abdullah, 2023).

Por otro lado, las restricciones blandas más comunes serían minimizar el número de exámenes consecutivos, repartir los exámenes de forma equitativa para cada estudiante, o la flexibilidad a la hora de trasladar un número determinado de exámenes sin necesidad de cambiar todo el calendario (Abdipoor, Yaakob, Goh, & Abdullah, 2023).

En el ETP, hay algunos términos que se podrían considerar como clave en lo que a la programación de exámenes respecta, ya que van a ser factores fundamentales a lo largo del TFG (Ceschia, Di Gaspero, & Schaerf, 2022):

- Examen. Para programar los exámenes, debe asignarse a cada uno de ellos un período y una sala específicos.
- Sesión de examen. Una sesión de examen consiste en todo el periodo durante el cual se realizan los exámenes o varios periodos durante un periodo de tiempo definido.
- Franja horaria/ Periodo. La sesión de examen abarca un determinado número de días, cada uno de los cuales se divide a su vez en distintas franjas horarias. Cada combinación de día y franja horaria designa un periodo específico dentro de la sesión.
- Aula. A cada sala se le asigna una capacidad determinada, que indica el número máximo de plazas disponibles. Algunos exámenes pueden requerir una o varias salas, mientras que otros pueden realizarse mediante una plataforma y no requieren espacios físicos.
- Recursos. Consideramos tres tipos principales de recursos: estudiantes, profesores y aulas, los eventos deben programarse teniendo en cuenta las restricciones de recursos, como las inscripciones de los estudiantes, las solicitudes de los profesores y la disponibilidad de las salas.
- Restricciones. Como es habitual, las restricciones se dividen en duras y blandas (las blandas también se denominan objetivos).

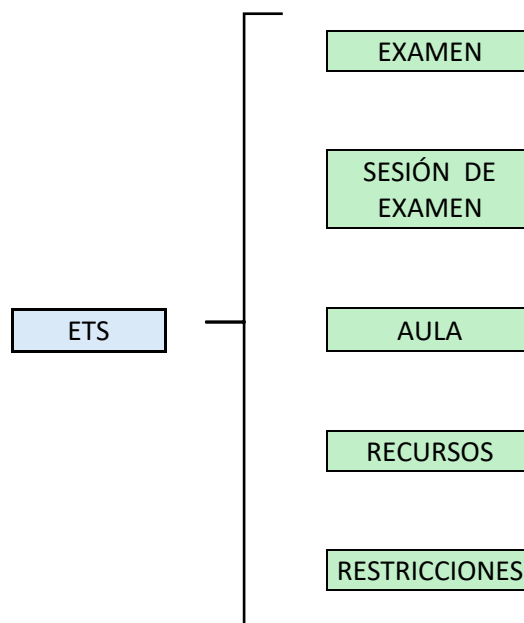


Figura 4. Esquema de los tipos de ETP. (Fuente: Elaboración propia)

Desde el punto de vista académico, podemos encontrar una clasificación de los problemas de programación de exámenes, dependiendo de la institución en la que se realiza la programación y el objeto de programación (Ceschia, Di Gaspero, & Schaerf, 2022):

- *High-School Timeyabling (HTT)*. Se trata del horario semanal de todas las clases de un instituto, evitando que los profesores se reúnan con dos clases al mismo tiempo y viceversa.
- *University Course Timetabling (CTT)*. Se aborda la programación semanal para todas las clases de un conjunto de cursos universitarios, evitando en la medida de lo posible el solapamiento de clases de cursos con estudiantes comunes.
- *University Examination Timetabling (ETT)*. Se encuentra la programación de los exámenes de un conjunto de cursos universitarios, evitando el solapamiento de los exámenes de los cursos que tienen estudiantes comunes, y repartiendo los exámenes para los estudiantes tanto como sea posible.

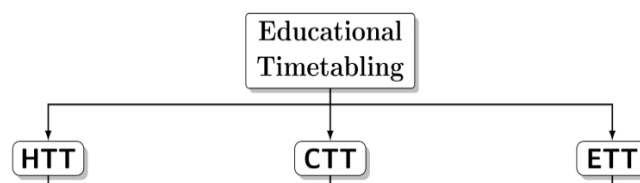


Figura 5. Esquema de los tipos de ETP. Fuente: (Ceschia, Di Gaspero, & Schaerf, 2022)

Además, encontramos diferentes clases también del Exam Timetable Problem (ETP) con diferentes bases de datos. Distinguiríamos entre Capacitados o No Capacitados (Siew, y otros, 2024).

- El **ETP no capacitado** no impone restricciones de capacidad de sala durante la programación y asume que se mantendrá el mismo nivel de capacidad en todas las salas.
- La **ETP con capacidad** implica tener en cuenta una capacidad limitada de aulas/plazas por período. En esta tipología, su vez, habría:
 - *ETP con capacidad de aula*: se impone una restricción de capacidad máxima que puede albergar cada sala en cada periodo.
 - *ETP con capacidad de franjas horarias*: se impone una restricción de capacidad a las aulas disponibles, especificando el máximo de asientos de la capacidad del aula en un periodo determinado.

3.3. Revisión de heurísticas para la programación del calendario de exámenes

En este apartado del capítulo de antecedentes, se va a abordar los estudios que tratan la aplicación de dicha base teórica de la investigación operativa en la programación de los exámenes, más conocido como Exam Timetabling Problems (ETP).

En los últimos 30 años se ha abordado la problemática de los exámenes debido a que se encuentra a la orden del día de las universidades del mundo moderno. Esta situación es de una gran complejidad y por ello no se han encontrado metodologías ciertas al 100% y que encuentren la mejor solución para este. Esto explica la creciente investigación sobre las heurísticas, ya que se trata de una metodología que proporciona una solución óptima sin ser la única verdadera.

3.3.1. Contextualización

Para la resolución de la problemática propuesta se ha decidido la utilización de heurísticas. En el día a día se producen problemas y, en el contexto de la Ingeniería de Organización, aparecen problemas de gran complejidad que no tiene una solución inmediata o para los que no existe un método preestablecido de resolución, es por esto que se requiere a la utilización de las heurísticas.

La palabra heurística viene del griego “eureka”, que significa hallar (Andrés Romano, 2018). Las heurísticas son: “procedimientos para resolver un problema matemático bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución” (Melian, Moreno Pérez, & Moreno Vega, 2003).

El resolver un modelo real, conlleva la creación de una versión más simple de la realidad, esta versión simplificada y abstracta se conoce como modelo. Para ello, se debe definir y estudiar el entorno en el que se va a realizar el problema, es por esto que se realizan una serie de pasos para hacer más fácil esta realidad (Vicens Salort, Ortiz Bas, & Guarch Bertolín, 1997):

1. **Definición.** Establecer el problema a resolver y lo que este requiere.
2. **Modelización.** Se basa en la creación de una versión simplificada de la realidad, clasificar el tipo de problemática a la que nos encontramos para poder abordarla de forma adecuada.
3. **Resolución.** Una vez modelado el sistema, podemos considerar por terminada la heurística cuando se encuentra una solución óptima y factible que responde a todos los requisitos propuestos al definir el problema.

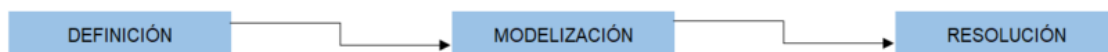


Figura 6. Esquema de los pasos para la modelización de una problemática (Fuente: Elaboración propia)

Cuando se utiliza una heurística, se encuentran generalmente un abanico diferente de soluciones, resultado de las diferentes iteraciones de la misma, por ello, al utilizar una heurística se puede asegurar que la solución es óptima pero no que es la mejor, ya que se tratan de problemas tan complejos que no se puede asegurar que es el único resultado cierto. Así, podemos encontrar soluciones que: 1. Son óptimas, 2. Cumplen las restricciones, pero no son óptimas, 3. No son óptimas. Según la referencia (Vicens Salort, Ortiz Bas, & Guarch Bertolín, 1997).

Como se ha comentado, se ha decidido utilizar una heurística de tipo constructiva, y un algoritmo de búsqueda local; sin embargo, no son los únicos tipos de heurísticas que existen. A continuación, se proponen las clases de heurísticas más destacadas, clasificadas debido a su diferenciación en la forma de encontrar la solución al problema (Andrés Romano, 2018):

- **Heurísticas Constructivas.** Consisten en la generación de la solución mediante un proceso diferenciado en pasos que elige elementos de un listado de candidatos que son candidatos a formar parte de la solución. Las heurísticas constructivas completan el proceso agregando paso a paso los elementos necesarios al construir paso a paso una solución completa del problema a resolver. Por lo tanto, se deben establecer los componentes que se pueden incluir en la solución y cómo se agregan.
- **Heurísticas basadas en la búsqueda de entornos.** Su exponente más sencillo es la mejora iterativa simple, cuyo funcionamiento se basa en la elección siempre de la primera solución que mejore la que ya teníamos y usarla como punto de partida para la siguiente mejora. En este tipo encontramos los algoritmos de búsqueda local.
- **Metaheurísticas.** Se trata de, en la manera aproximada, de todos los algoritmos en búsqueda de propósito general. En el proceso de encontrar la solución óptima, estas abordan evitando la convergencia demasiado temprana mediante el equilibrio entre dos aspectos contradictorios de la resolución: la intensificación y la diversificación. Destacan diferentes tipos como el GARSP (Greedy Randomized Adaptive Search Procedure), la búsqueda local iterativa, Algoritmo de Recocido Simulado o el Algoritmo genético.

3.3.2. Heurísticas Avanzadas aplicadas al ETP

Una vez abordado la contextualización de las heurísticas, en esta sección se pretende tratar como dar solución a la problemática de la programación de exámenes mediante herramientas de investigación operativa de diferentes tipos.

Primero, cabe destacar que se puede abordar la solución de este problema desde la optimización matemática con métodos matemáticos, categorizando la programación lineal o no lineal como sus diferentes subtipos, aunque este tipo de herramientas ha demostrado no ser altamente eficiente para problemas NP como el que se aborda en este TFG. Por otro lado, encontramos las heurísticas, que comprende heurísticas constructivas, grafos coloreados o la descomposición.

Si lo abordamos desde el punto de vista heurístico, encontramos también (Siew, y otros, 2024):

- **Metaheurísticas:** metodologías de alto nivel que combinan heurística subyacente, empleando estrategias inteligentes para explorar y explotar el espacio de búsqueda y guiar simultáneamente los procedimientos heurísticos con el aprendizaje técnicas. La clasificación de una metaheurística como solución única o basada en la población es contingente sobre el número de soluciones examinadas en cada iteración del procedimiento de exploración.

- **Matheurística (MAH):** hibrida metaheurísticas con técnicas de programación matemática, utilizando modelos de programación temática en un marco heurístico.
- **Hiperheurística (HH):** explora un espacio de búsqueda heurística de bajo nivel o componentes heurísticos, delineando dos tipos principales:
 - *Hiperheurísticas de generación:* crear nuevas heurísticas mediante componentes y operadores.
 - *Hiperheurística de selección:* mejorar las soluciones mediante la selección de heurísticas de bajo nivel de un conjunto proporcionado.

Por último, tenemos también las Aproximaciones Híbridas, que tienen un alto nivel de integración ya sea incorporando algoritmos de Metaheurística en una herramienta heurística (Siew, y otros, 2024).

3.3.2.1. Heurística directa

El coloreado de gráficos es un tema que se ha estudiado desde principios de la década de 1970. La coloración de gráficos considera un caso especial de etiquetado de gráficos, eso se define como una asignación de etiquetas llamadas "colores" a elementos. La coloración se ha utilizado en muchas aplicaciones y se ha utilizado ampliamente para Problemas de programación y, en su forma más simple, el método se basa en colorear los vértices de un grafo de tal manera que no haya dos adyacentes (Gashgari, y otros, 2018).

La metodología consta de dos fases, primero se emplea un algoritmo heurístico basado en la coloración de grafos que se utiliza para dividir los exámenes en grupos mutuamente independientes y, la segunda fase, consiste en combinar este algoritmo con otro Algoritmo basado en la lista de cursos para asignar exámenes en las habitaciones.

3.3.2.2. Algoritmos genéticos

El Algoritmo Genético es un método para resolver un problema de optimización creado por John Holland a principios de la década de 1970. El objetivo es encontrar la solución que se ajuste a los criterios (restricciones). El primer paso en el Algoritmo Genético es comenzar con un conjunto de diferentes diseños de soluciones factibles para la programación de horarios referencia (Gashgari, y otros, 2018).

El grupo de soluciones representará a una población y una sola solución a menudo se denomina cromosoma, cada uno de ellos es factible. La solución tiene diferentes rasgos que pueden mutar aleatoriamente, esto da como resultado tener una nueva población. Después de eso, la selección de la mejor solución el proceso ocurre donde la solución más adecuada es elegida (Gashgari, y otros, 2018).

Son muchas las universidades que aplican esta técnica, por ejemplo, por ejemplo, la Facultad de Ingeniería Eléctrica y Computación en la Universidad de Zagreb implementó con éxito el Solución basada en algoritmos genéticos (Gashgari, y otros, 2018).

3.3.2.3. *Búsqueda tabú*

La Búsqueda Tabú es una meta-heurística basada en el trabajo propuesto por Fred Glover diseñado para resolver la optimización combinatoria problemas. En el caso de que se realice una búsqueda exhaustiva a través de un espacio de solución no es factible, la Búsqueda Tabú ofrece una forma de buscar en un espacio de soluciones de manera eficiente (Gashgari, y otros, 2018).

Una limitación de otras estrategias especializadas como métodos decentes y escalada de colinas heurísticas que están diseñadas para resolver problemas combinatorios es que producen soluciones óptimas locales que a menudo no son óptimos globales. La Búsqueda Tabú tiene la capacidad de trabajar con estas otras estrategias de resolución de problemas para guiarlos más allá de los límites de la optimalidad. El algoritmo de Búsqueda de Tabú se ha adaptado a los años para resolver el problema del calendario de exámenes (Gashgari, y otros, 2018).

El algoritmo TS fue aplicado por Álvarez-Valdés, Crespo y Tamarit (1997) en el problema de programación académica para generar un horario maestro, un horario que no tiene en cuenta las preferencias de los estudiantes. Los autores emplearon 2 fases consecutivas que comprendían un método principal utilizado para generar un horario factible sin conflictos y, posteriormente, un algoritmo de búsqueda tabú para para mejorar el horario generado. Además, en el estudio se añadieron otra fase para mejorar el proceso de asignación de salas hacia el final de la fase 2. El objetivo de mejorar la asignación de aulas era minimizar el movimiento de los estudiantes entre clases, que es un criterio importante para aumentar la calidad del horario. Para diversificar la estructura de vecindad, los autores emplearon 3 métodos de movimiento: movimiento simple, intercambio, intercambio múltiple y oscilación de pesos, lo que permitió una mayor exploración y de los tres métodos multiswap demostró ser el más fiable, ya que permitía realizar importantes modificaciones en la solución (Wibowo, Teoh, & Ngadiman, 2013).

También se hicieron comparaciones con los parámetros de lista de candidatos, lista tabú y recuperación de soluciones y se descubrió que el uso de una lista de candidatos reduce el espacio de búsqueda en torno a la vecindad potencial de la solución candidata. Utilizando la mejor solución candidata, se llevó a cabo otro proceso de búsqueda alrededor de la región para ver si se podía obtener una solución mejor. Con los resultados obtenidos se observó que los métodos de movimiento e intercambio simples no eran apropiados para resolver un problema relativamente grande. La tasa de utilización de cada aula de la institución arrojó una media del 83 %, lo que es considerablemente satisfactorio (Wibowo, Teoh, & Ngadiman, 2013).

El autor llegó a la conclusión de que las combinaciones ideales para el algoritmo TS deberían consistir en las siguientes (Wibowo, Teoh, & Ngadiman, 2013):

1. El movimiento es multiswap (el parámetro más crítico y es ideal para explorar complejos vecindario).
2. Uso temporal de la lista de candidatos para mejorar la función objetivo y reducir el número de movimientos de búsqueda.
3. Lista Tabú con longitud que cambia dinámicamente (mejora significativamente la robustez del algoritmo).
4. Oscilación estratégica de pesos (para diversificar la exploración del espacio de búsqueda).

5. Recuperación (Recovery) de la mejor solución conocida tras un número determinado de iteraciones sin mejora (un proceso de intensificación para obtener mejores soluciones).

3.3.2.4. Heurísticas constructivas

Las heurísticas de construcción se basan en la búsqueda de la solución a través de una metodología paso a paso por la que partiendo de una lista de proposiciones a solución se eligen estos para que formen parte de la solución final (Andrés Romano, 2018).

Para comenzar, se ha de elegir de forma adecuada la forma de representación de la solución y, por otro lado, se ha de tener cuenta la definición y el proceso de elección de los candidatos que se incluyen en la solución final al terminar cada una de las iteraciones. Además, otro factor importante es la forma en la que se eligen dichos candidatos. Cabe destacar el concepto de solución vacía, que equivale a no haber añadido ningún candidato a la selección final (Andrés Romano, 2018).

Los elementos del listado que son considerados aptos para formar parte de la selección se denominan candidatos y la manera de añadirlos se fundamenta en un concepto conocido como criterio constructivo o regla de prioridad de los candidatos. La manera de evaluar si son aptos o no los candidatos para formar parte de la solución final se denomina función greedy o voraz, esta función es un criterio que ha de ser sencillo de obtener y que ha de tener en consideración el objetivo final a (Andrés Romano, 2018).

Cuando comienza el proceso, una vez añadido el elemento número 1 a la solución, comienzo un bucle de iteraciones que es consecutivo hasta que se llega a la solución completa, véase en el caso objeto del TFG, la asignación de todas las asignaturas del listado disponible de la ETSII. El bucle coge un nuevo elemento del listado y lo va añadiendo a la solución final.

De forma más esquemática, siendo W el conjunto de elementos que pueden pertenecer a la solución, t un índice para identificar el elemento actual que se acaba de incorporar a la solución en curso, y f_{ti} es la función greedy del elemento i con respecto del elemento t , la estructura de una heurística constructiva sería (Andrés Romano, 2018):

1. *Inicialización*

Seleccionar un elemento i entre los candidatos posibles y añadirlo a la solución.

$T=i$ y descontar del conjunto W

2. *Repetir mientras haya elementos por añadir a la solución*

Elegir i de W de tal manera que i tenga el menor valor de $f_{ti} \forall i \in W$.

Añadir i a la solución usando un criterio constructivo.

$T=i$ y descontar i del conjunto W .

Las heurísticas constructivas se utilizan ampliamente para la resolución de la programación de exámenes. Como ejemplo, se va a abordar un caso aplicado de la Universidad de Malasia Pahang (UMP), también conocida como Kolej Univerisiti Kejuruteraan dan Teknologi Maysia (KUKTEM), tratado en el artículo (Kahar & Kendall, 2014).

Esta universidad es consistente en cinco facultades con 3.550 estudiantes con un total de 17 cursos ofertados. Para describir el problema se abordan las restricciones duras más destacables del problema:

1. Ningún estudiante puede estar en dos exámenes simultáneamente.
2. El número total de estudiantes asignados a un aula en particular tiene que ser menos que el total de la capacidad de la misma.
3. Solo un examen es establecido por aula. Cabe decir que se pueden combinar el mismo examen, aunque sea de distintos grados.

Debido a que las clases tiene un máximo de 100 alumnos, en la realización de los exámenes se han de separar en diferentes aulas.

La base de datos en la que se basa el estudio es la siguiente:

Constraints		Toronto	Nottingham	Melbourne	UKM	UITM	ITC2007	UMP
Examinations	Clash free	Hard	Hard	Hard	Hard	Hard	Hard	Hard
	Scheduled all exams		Soft	Soft	Hard	Hard		Hard
	Weekend scheduled					Soft		
	Exam preference <ul style="list-style-type: none"> ▪ specified arrangement: <i>sa</i> ▪ specified room: <i>sr</i> ▪ large exam schedule first: <i>lf</i> ▪ restriction on exam in particular timeslot: <i>rt</i> ▪ scheduled combined exam in the same timeslot: <i>ct</i> 			Hard (<i>rt</i>)		Hard (<i>ct</i>)	Hard (<i>sa</i>) Soft (<i>lf</i>)	
	Consecutive exam <ul style="list-style-type: none"> ▪ two exam in a row: <i>2r</i> ▪ two exam in a day: <i>2d</i> ▪ two exam in a row overnight: <i>2n</i> ▪ three exam in a day: <i>3d</i> 		Soft (<i>2d</i> & <i>2n</i>)	Soft (<i>2d</i> & <i>2n</i>)	Hard (<i>3d</i>) Soft (<i>2r</i>)		Soft (<i>2r</i> and <i>2d</i>)	
Timeslot related	Timeslot preference <ul style="list-style-type: none"> ▪ minimise/avoid usage: <i>tu</i> 						Soft (<i>tu</i>)	
	Timeslot length <ul style="list-style-type: none"> ▪ mixed duration of exams in one timeslot: <i>mt</i> 						Hard Soft (<i>mt</i>)	
	Spreading <ul style="list-style-type: none"> ▪ specified spread: <i>ss</i> 	Soft	Hard (<i>ss</i>)	Soft	Soft	Soft	Soft (<i>ss</i>)	Soft
Rooms related	Room distance							Soft
	No sharing of room with other exams <ul style="list-style-type: none"> - for specified exam only: <i>se</i> 				Hard (<i>se</i>)			Hard
	Room preference <ul style="list-style-type: none"> - consecutive exam scheduled in the same room: <i>cr</i> - minimise/avoid usage: <i>ru</i> - specified room: <i>sr</i> 				Hard (<i>cr</i>)		Hard (<i>sr</i>) Soft (<i>ru</i>)	
	Split exam into different rooms <ul style="list-style-type: none"> ▪ same building only: <i>sb</i> ▪ as close as possible: <i>cp</i> 							Hard (<i>sb</i>) Soft (<i>cp</i>)
	Capacity <ul style="list-style-type: none"> ▪ total seats: <i>ts</i> ▪ individual room: <i>ir</i> 		Hard (<i>ts</i>)	Hard (<i>ts</i>)	Hard (<i>ts</i> and <i>ir</i>)		Hard (<i>ir</i>)	Hard (<i>ir</i>)

Figura 7. Resumen de la base de datos del caso de la UMP. Fuente: (Kahar & Kendall, 2014).

En la heurística propuesta por el autor, el conjunto de datos se tomó de la Universidad de Malasia Pahang (UMP) para el semestre 1 de 2007. El número total de exámenes es de 252, en todos los 17 programas ofrecidos por 5 facultades, sin embargo, debido al requisito de exámenes combinados, el conjunto de datos se ha procesado previamente y a los exámenes combinados se les asigna un nuevo código de examen y es tratado como un examen grande. Esto da como resultado un total de 157 exámenes. Al combinar estos exámenes, ayuda a minimizar y optimizar el uso de las habitaciones. El número total de estudiantes es de 3550 con 12.731 matrículas.

El espacio total disponible para este conjunto de datos es de 24 salas, y cada aula tiene una capacidad determinada. Para ayudar a nuestra heurística constructiva en el proceso de búsqueda de la(s) habitación(es) más adecuada(s) y minimizando el valor del costo relacionado con la habitación, generamos una lista de aulas agrupadas (basadas en la lista de aulas proporcionadas). Estas agrupaciones de habitaciones predeterminadas se generan dentro del mismo edificio solamente.

Hay que tener en cuenta que limitamos las agrupaciones de aulas hasta un máximo de 4 salas posibles para cada examen. En sus observaciones, 4 habitaciones son adecuadas para satisfacer cualquier capacidad de examen. Además, el aumento de las posibilidades de agrupación de aulas (>4) obviamente aumentarán el costo de la distancia, dividiendo el costo y el espacio de búsqueda. Las agrupaciones de aulas se ordenan en orden decreciente en función de la capacidad total de las habitaciones. Al hacerlo, se puede buscar directamente la(s) habitación(es) adecuada(s) y terminar el procedimiento de búsqueda cuando se detecta una capacidad de sala inadecuada Encontrado.

Para ilustrar el procedimiento, los autores proporcionan el siguiente ejemplo: “Supongamos que tenemos cinco habitaciones en dos edificios diferentes, donde cuatro de las salas están en el mismo edificio, y cada habitación tiene una capacidad. El coste de viaje para habitaciones en diferentes edificios no se muestra, ya que esto no está permitido, por lo tanto, podríamos crear 15 agrupaciones de habitaciones con 14 agrupaciones de habitaciones de los edificios W y 1 grupo de aulas del edificio X. Cada una de las agrupaciones de aulas tienen su nueva capacidad total, el coste de la distancia (total del valor de distancia anterior a la matriz de distancia para cada habitación) y el coste de división. Estas agrupaciones de habitaciones se ordenan en función de en su capacidad. Tener el orden decreciente de las agrupaciones de aulas predeterminadas ayuda al algoritmo de búsqueda a seleccionar las aulas más adecuadas, con el objetivo de minimizar el valor de los costes relacionados con el aula y acelerar la búsqueda deteniendo el procedimiento de búsqueda de aulas cuando se alcanza la capacidad de la agrupación de estas 4 aulas”.

Los experimentos se llevan a cabo utilizando enfoques heurísticos de grafos que incluyen el grado más grande (LD), el grado ponderado más grande (LWD), grado de saturación (DE) y mayor matrícula (LE). El A continuación se presenta una descripción de estos métodos:

- Grado con mayor número de matriculados (LD): esta heurística toma los exámenes que tienen la mayor cantidad de matriculados con otros exámenes y se les programa primero.
- Grado ponderado más grande (LWD): esta heurística es similar a la anterior, excepto que toma los exámenes que tienen el mayor número de estudiantes que están involucrados en el conflicto y los horarios ellos primero.
- Matrícula más grande (LE): esta heurística realiza exámenes con el mayor número de alumnos inscritos y los horarios Primero.
- Grado de saturación (DE): esta heurística elige los exámenes que tener el menor número de períodos disponibles en el calendario que se pueden seleccionar y programarlos primero.

3.3.2.5. Algoritmos de búsqueda local

Los algoritmos o heurísticas de Búsqueda Local o Búsqueda de entornos surgen de los problemas de optimización con un gran tamaño del espacio de soluciones S que contiene a todas las soluciones posibles de la problemática en cuestión. El espacio de búsqueda S' puede ser menor o igual al espacio de soluciones primitiva y, ahí aparecen los algoritmos de búsqueda local (Andrés Romano, 2018).

Cuando se pretende el diseño de una heurística de esta tipología se ha de establecer de forma concisa los siguientes elementos (Andrés Romano, 2018):

- La representación de las soluciones y del espacio de soluciones.
- Definición de la solución de partida.
- Un operador de generación de vecinos y fijado el operador se define el entorno de una solución.

Tomar un punto inicial del proceso de búsqueda es uno de los factores fundamentales de esta metodología para su buen funcionamiento, por ello si se es sabedor de una solución preestablecida que es buena, es fundamental tomarla como punto de partida, en nuestro caso, la solución obtenida de la heurística constructiva.

En el caso de este TFG, se va a implantar la clase de Algoritmo de Búsqueda Local más conocida, la mejora iterativa simple, es su versión más sencilla, y su procedimiento es el siguiente (Andrés Romano, 2018):

1. *Inicialización*

Seleccionar una solución inicial x'

Definir la mejor función objetivo $f(x')$

2. *Repetir mientras no se cumpla un criterio de parada*

Aplicar una transformación de la solución x' , a esta transformación se denomina x

Si $f(x) < f(x')$ entonces $x' = x$ y se guarda la mejor función objetivo $= f(x)$

Si no se cumple la condición de parada se vuelve al paso 2

3. *Mostrar x' y la mejor función objetivo*

La búsqueda local iterada (ILS) es una metodología relativamente sencilla que ha tenido éxito en diversos ámbitos. Funciona de forma iterativa alternando entre la aplicación de un operador de movimiento a una solución y realizar una búsqueda local en la solución perturbada. Este principio de búsqueda se ha utilizado varias veces en diferentes comunidades de investigación y con distintos nombres. (Soria-Alcaraz, Özcan, Swan, Kendall, & Carpio, 2016).

El algoritmo de búsqueda local iterado comienza con una solución inicial e itera varias veces para crear una solución candidata basado en el existente. Si la solución candidata encontró que ser mejor que la existente, la solución existente será actualizado con el candidato. Continúa el procedimiento de búsqueda hasta que no se haya producido ninguna mejora de la solución durante varios Iteraciones (Soria-Alcaraz, Özcan, Swan, Kendall, & Carpio, 2016).

En el caso aplicado a la problemática de la programación de exámenes, en las investigaciones realizada en este campo para poder dar una solución, se toma como punto de partida una solución ya dada y, en base a ella, se realizan permutaciones en este horario solución ya dado y que pretenden mejorar la primera. Encontramos dos partes para aplicar el algoritmo seleccionado (Ekanayake, Subasinghe, Ragel, Gamage, & Attanayaka, 2019):

- Generar Solución Inicial en ILS. Para empezar, creando la solución inicial, un evento aleatorio (E) del conjunto de eventos no asignados (E') se selecciona y es añadido al conjunto creado temporalmente, que es inicialmente un conjunto vacío. Esto se repite hasta que no se pueda agregar ningún evento a E sin causar un choque de estudiantes con cualquier evento que ya esté en los eventos seleccionados. A continuación, se toma el intervalo de tiempo (t) y cada evento se asignará a una sala que se encuentre dentro de la franja horaria seleccionada (T). Los eventos que no pueden coincidir con un aula se eliminan de la lista de eventos (Et) y se vuelven a agregar a E. La solución inicial se obtiene después de que se hayan tomado todos los intervalos de tiempo en consideración para el proceso anterior.
- Perturbación de mejora. Una vez obtenida la solución inicial, la búsqueda local iterada (ILS) utiliza un mecanismo de perturbación de mejora para mejorar aún más la solución existente. Para lograr esto, se realiza el intercambio y la nueva solución se comprueba con el comando existente para la viabilidad con el uso de violaciones de la Restricciones blandas.

4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS

4.1. Introducción

El capítulo 4 consistirá en la presentación de la herramienta heurística de tipo constructiva que se propone para la resolución de la problemática de la programación de exámenes de un cuatrimestre de una Escuela o Facultad, en concreto para el intento de mejora de la programación de la Escuela Técnica Superior de Ingeniería Industrial de Universidad Politécnica de Valencia. En una primera instancia, se define la situación que plantea la problemática objeto de tratar. De forma seguida, se propone la herramienta heurística para la resolución del problema, es decir el modelado. Por último, se concluye con una breve explicación de la arquitectura utilizada para la implementación de la heurística y su resolución.

4.2. Hipótesis de partida

Antes de comenzar con la descripción de cómo se ha abordado la problemática, cabe destacar que se han tenido en cuenta las siguientes hipótesis:

1. Se debe asignar un día y turno para la realización de su examen a cada asignatura impartida en un mismo cuatrimestre.
2. No se puede asignar asignaturas de la misma titulación y curso en el mismo día, salvo si son optativas; es decir, si ya se ha programado para un turno del día el examen de una asignatura, el resto de las asignaturas de la misma titulación y curso han de ser asignadas en un día distinto.
3. Se debe asignar al menos un aula a cada asignatura. La capacidad de examen de las aulas asignadas a cada asignatura debe ser igual o superar el número de matriculados de dicha asignatura.
4. Una vez asignada a un día, turno y aula, la asignatura queda bloqueada, al igual que el aula asignada en ese turno. Es decir, que no se puede volver a programar un examen de dicha asignatura, ni volver a asignar el aula para ese mismo día y turno.
5. Todas las aulas de la Escuela se encuentran disponibles a la hora de realizar la asignación de los exámenes de las aulas. No obstante, la herramienta desarrollada permite indicar si alguna de ellas estuviera ocupada inicialmente para no considerarla.
6. No se programan los exámenes para las asignaturas de idiomas, ya que se ha supuesto que su asignación no corresponde directamente a la ETSII.
7. Las asignaturas compartidas por el grado de GIOI y el doble grado de GIOI+CD se ha tomado como una única, agrupando el número de matriculados de ambas titulaciones.
8. Debido a la reciente incorporación de la titulación, se ha supuesto que todas las asignaturas del doble grado de GIOI+CD contará con 30 matriculados.
9. Se asume que los exámenes de las asignaturas correspondientes a cursos alternos deben programarse en turnos contrarios para asegurar la compatibilidad de los exámenes.

10. Se identifican diferentes vías o ramas de especialización en las titulaciones para asegurar que sus exámenes sean compatibles (no se programen en el mismo turno y día) pero que los exámenes correspondientes al mismo curso y diferente vía puedan solaparse. Por ejemplo, todos los exámenes correspondientes a asignaturas de la especialidad de Ingeniería de Organización en el MUJI deben ser compatibles entre sí, pero no con los exámenes de las asignaturas de la especialidad de Energía Eléctrica u otras.
11. Las distancias entre clases han sido estimadas con pasos como unidad de medida. Se considera que las distancias entre aquellas clases que se encuentran conectadas por una puerta interna son de cero.

4.2. Definición de la problemática

La distribución de los cursos de grados y de postgrado en una Escuela consiste en que, como se ha comentado en el Capítulo 2. DESCRIPCIÓN DE LA PROBLEMÁTICA, cada grado cuenta con cuatro cursos (salvo el reciente doble grado que cuenta con 5) y los másteres con 1 curso (salvo el máster habilitante a ingeniero industrial que cuenta con 2 cursos), divididos en dos cuatrimestres A y B. En cada asignatura de cada grado se realiza una prueba a mediados del cuatrimestre y otra al final, contando con un examen global en periodo extraordinario. Además, hay asignaturas que, bajo el mismo nombre, o similar, imparten el mismo contenido y emplean el mismo examen para evaluar a sus estudiantes, aunque en diferentes carreras o máster, a este tipo se les denomina como asignaturas equivalentes.

La estructura del horario de exámenes viene regida por una serie de normas, comenzando con que el primer curso de cada uno de los grados y másteres se asignará en un turno, y el segundo curso en el contrario, y así sucesivamente, quedando el tercer curso encuadrado en el turno del primer curso y el cuarto curso en el del segundo curso. La asignación es alternativa para cada cuatrimestre, lo que significa que los cursos que estuvieran asignados al turno de mañanas en el cuatrimestre A, pasan a ser de tardes en el cuatrimestre B. Esta norma encuentra su raíz en intentar la minimización de los solapes de exámenes para alumnos que se encuentran cursando asignaturas de diferentes cursos, haciendo así que, si por ejemplo un alumno se encuentra en el tercer curso, pero sigue matriculado en una asignatura de segundo, pueda examinarse a la vez de las dos sin que se dé la situación que le coincidan en el mismo “slot”.

Además de establecer horarios de mañana o tarde, se debe especificar la fecha en que se llevará a cabo el examen, así como las aulas. En este caso, es necesario asegurarse de que la capacidad total máxima de las aulas asignadas a un espacio sea mayor o igual a la cantidad de estudiantes inscritos en cada asignatura. Siempre se intenta asignar aulas para un mismo examen en el mismo edificio y planta para facilitar la vigilancia de los exámenes por parte de los profesores, evitar equivocaciones de los estudiantes y simplificar la resolución de dudas durante el examen. Además, es crucial garantizar que haya al menos tantos profesores de vigilancia como aulas asignadas a un examen. A pesar de que este aspecto generalmente se organiza en función de la asignatura, la Unidad Docente o el Departamento que la imparte.

Por último, pero no menos importante, es fundamental siempre intentar reducir la cantidad de aulas que se utilizan en los exámenes. La disminución del número de aulas empleadas reduce el gasto de energía de la universidad y reduce la ineficiencia en el uso de los recursos humanos, especialmente el profesorado y el equipo de mantenimiento y limpieza.

El objetivo del capítulo es resolver el problema planteado creando una heurística de tipo constructivo. Se debe establecer una fecha y un horario para cada examen durante el período de exámenes regulares, con el fin de evitar complicaciones para los estudiantes y tratar de mantener lo más posible la distancia entre aulas. Además, se deben asignar las aulas necesarias y los profesores de supervisión a cada examen programado.

4.3. Modelo de Heurística Constructiva para la programación de exámenes considerando la distancia entre aulas

4.3.1. Base de datos y Restricciones

Para la realización de la heurística, primero se ha realizado una recopilación de datos que será fundamental para el correcto funcionamiento de la misma. Los datos que han sido recopilados se pueden dividir en cuatro bloques: las asignaturas con sus respectivos cursos, cuatrimestres, grados; las asignaturas y sus respectivos equivalentes de otros grados; las clases con sus capacidades; y las distancias entre las mismas.

En una primera instancia, se encuentran los diferentes grados, que tienen un código propio y su abreviatura. Estos grados los subdividimos en los diferentes cursos que se realizan en los respectivos grados y, además, se realiza una subdivisión dependiendo de la rama o vía escogida. Por último, ya encontramos las asignaturas, divididas en sus respectivos cuatrimestres, cada una de ellas tiene su código propio y un número total de matriculados que cursan dicha asignatura.

Todos los datos a los que se hace referencia se ven ejemplificados en las figuras 9 y 10 de la página de a continuación.

Por otro lado, las asignaturas tienen sus equivalentes por lo que se ha tenido en cuenta y se ha tabulado estas relaciones. En este caso se ha tabulado el nombre genérico de la asignatura, los códigos a los que hace referencia, y el nombre, código y abreviatura de la titulación, curso, semestre y número de matriculados.

CÓDIGO TITULACIÓN	CURSO	SEMESTRE	CÓDIGO ASIGNATURA	Nº MATRICULADOS	VIA
156	1	1	14004	15	0
156	1	1	14002	15	0
156	1	2	13998	15	0
156	1	2	14000	15	0
156	1	2	14005	15	0
156	1	2	14003	15	0
156	1	2	14019	15	0
156	2	2	14015	15	0
156	2	2	14026	15	0
156	2	2	14008	15	0
156	2	2	14001	15	0
156	2	2	14010	15	0
156	2	2	14006	15	0
156	3	1	14013	15	0
156	3	1	14016	15	0
156	3	1	14011	15	0
156	3	1	14020	15	0

Figura 8. Ejemplo de tabla de datos de las asignaturas (Fuente: Elaboración propia)

Assignatura	Assignatura
11420	11531
11421	11524
11422	11532
11423	11528
11424	11525
11425	11527
11426	11529
11427	11534
11428	11526
11429	11536
11430	11530
11431	11533
11432	11535
13232	14270
13233	14269
13235	14664
13235	14667
13236	12977
13756	13751
13757	13752
13757	14060
14413	14414

Figura 9. Ejemplo de tabla de datos de las asignaturas Equivalentes (Fuente: Elaboración propia)

En cuanto a las aulas, se ven reflejadas como datos: el número de alumnos que pueden albergar (tanto a la hora de impartir las clases como cuando se realicen exámenes) ya que en ese caso la capacidad se ve reducida a la mitad. Además, se ha medido en pasos la distancia entre las aulas, de esta forma se podrá tener en cuenta a la hora de la asignación y se ha tabulado la relación entre las aulas con su respectiva distancia entre ellos.

AULA	CAPACIDAD NORMAL	CAPACIDAD EXAMEN
11	148	74
12	104	52
13	104	52
14	104	52
21	148	74
22	100	50
23	100	50
24	96	48
25	100	50
31	148	74
32	100	50
33	100	50
34	100	50
35	100	50
41	148	74
110	104	52
111	56	28
112	84	42

Figura 10. Ejemplo de tabla de datos de las aulas (Fuente: Elaboración propia)

AULA 1	AULA 2	DISTANCIA
11	12	0
11	13	7
11	14	30
11	21	57
11	22	60
11	23	63
11	24	89
11	25	92
11	31	118
11	32	128
11	33	131
11	34	153
11	35	156
11	41	179
11	110	330
11	111	330
11	112	351
11	211	351
11	212	351
11	213	281
11	214	301

Figura 11. Ejemplo de tabla de datos de la distancia entre las aulas (Fuente: Elaboración propia)

A la hora de aplicar la heurística cabe resaltar que se han tenido en cuenta las siguientes restricciones de tipo duras:

1. No puede haber dos exámenes de la misma titulación, curso y vía en el mismo turno a menos que una de ellas sea una optativa.
2. Dos exámenes de cursos sucesivos pueden estar el mismo día, pero no en el mismo turno.
3. Si ya se ha asignado en un turno del día una asignatura, las asignaturas de su mismo grado y curso han de ser asignadas en un día distinto.
4. Las asignaturas de los cursos impares son designadas en un turno.
5. Las asignaturas de los cursos pares son designadas en el turno contrario.

Las restricciones de tipo blandas serían:

1. La distancia entre las aulas asignadas a una misma asignatura ha de ser 0. Esta condición no siempre se cumple, pero es un objetivo a intentar cumplir, por eso es considerada blanda.
2. Las aulas designadas a una asignatura deben de encontrarse en el mismo edificio.

4.3.2. Criterio o Regla de selección

Como se trata de una heurística constructiva, se pretende comenzar su desarrollo explicando la regla de decisión que decide la programación de los diferentes exámenes en las aulas disponibles y en los horarios determinados.

El criterio de la heurística propuesta se basa en dos objetivos que definen las reglas de decisión aplicadas durante todo el algoritmo. En una primera instancia, se toma como objetivo principal la reducción de las distancias entre las aulas designadas a una misma asignatura, es por esto que en el momento de la asignación de las aulas se tiene como criterio de selección de las aulas, la elección del aula se encuentre más cercana a la primera candidata.

En segundo lugar, en el caso de que un examen, debido a que su número de matriculados es inferior a la capacidad del aula más grande disponible, es posible que sea asignado en una única aula, se sigue el criterio de elegir la candidata cuya capacidad mejor se ajuste al número de matriculados de la asignatura del examen. Esto quiere decir que se escogerá el aula que minimice el valor de la holgura entre la capacidad del aula candidata y el número de matriculados de la asignatura a designar.

4.3.3. Algoritmo paso a paso

4.3.3.1. Inicialización

Como se puede apreciar en la figura 12, la heurística comienza con la lectura del cuatrimestre (A o B) y turno (mañana, tarde o ambos) para el que el usuario desea realizar la programación de exámenes y el número de días lectivos con el que contará periodo de exámenes.

Por otra parte, se realiza la creación y lectura de las matrices necesarias para el correcto funcionamiento de la heurística. Para ello se ha creado una función llamada “*Calcular_Tamaño*” encargada de, conociendo la Hoja en la que se guardan los datos de entrada, identificar el tamaño con el que contará la posterior matriz. También se ha creado una función llamada “*Leer_Tabla_Col*” encargada de, una vez conocido el tamaño de la matriz, redimensionarla y guardar en ella todos los datos de entrada disponibles en el fichero de origen. Las matrices creadas que siguen este método son:

- “*Datos_Asignaturas_Cod*”. Matriz que cuenta con seis columnas y con tantas filas como número de asignaturas impartidas en la ETSII. En la primera columna se recoge la titulación a la que pertenece la asignatura, en la segunda columna el curso, en la tercera el cuatrimestre, en la cuarta el código identificador de la asignatura, en el cuarto el número de estudiantes matriculados y en la quinta la especialidad o vía a la que pertenece la asignatura.
- “*Cod_Asig_Eq*”. Matriz que cuenta con dos columnas y tantas filas como combinaciones existan entre las asignaturas que son equivalentes. En la primera columna se encuentra la asignatura objetivo y en la segunda su equivalente.
- “*Datos_Capacidad*”. Matriz que cuenta con tres columnas y tantas filas como aulas existen en la ETSII. En la primera columna encontramos la codificación de las aulas, en la siguiente columna la capacidad para albergar clases y, en tercer lugar, su capacidad a la hora de la realización de un examen.
- “*Datos_Distancias*”. Matriz que cuenta con tres columnas y tantas filas como combinaciones existan entre las aulas de la ETSII. En la primera columna se encuentra el aula origen (desde la que se mide la distancia), en segundo lugar, el aula destino (hasta la que se mide la distancia) y en la tercera columna la distancia entre estas.
- “*Optativas*”. Matriz que recoge las optativas con su código.

Con una función parecida “*Leer_Disponibilidad*” se lee la disponibilidad de las aulas calificadas como 1 si se encuentra ocupada o 0 si está libre y se guarda la información en la matriz Disponibilidad, matriz con tantas columnas como días más uno (ya que en la primera columna se guarda el nombre del aula) se deseen programar y tantas filas como turnos se vayan a programar, si es un solo turno el número de filas será el número de aulas, sino el doble de las aulas que haya en la ETSII, debido a que hay dos turnos. A continuación, siguiendo el criterio de crear la matriz, se genera la matriz solución, en base al número de aulas y a los días deseados a programar, introducidos por el usuario.

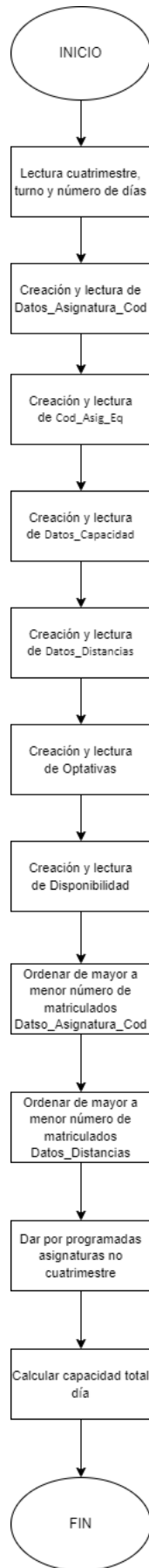


Figura 12. Diagrama de flujo de Inicialización (Fuente: Elaboración propia)

Al comenzar la heurística, como primera regla se ordena de mayor a menor el número de matriculados de todas las asignaturas de la escuela, independientemente del grado, curso, vía o cuatrimestre al que pertenecen. La asignatura que se encuentra en la cima de dicho listado es la que se elige como primera asignatura candidata para ser programada. La asignatura que se encuentra en la parte superior de ese listado es la que se elige como la primera asignatura candidata para ser asignada en el salón de clases disponible el día designado. Por otra parte, se repasan todas las asignaturas para identificar cuáles de ellas corresponden al cuatrimestre para el que se quieren programar los exámenes. En el caso de que la asignatura se imparta en otro cuatrimestre, se indica que está ya ha sido programada en el vector “asignada”, este vector tiene tantas casillas como asignaturas hay y, en él, se indica con un 1 cuando esa asignatura ya está programada o no se debe programar y con un 0 cuando se encuentra pendiente de programar.

Tras esto, se aplica una nueva regla que consiste en el ordenado nuevamente de mayor a menor pero esta vez de las aulas con respecto de su capacidad. Así, el aula ubicada en la primera posición de la matriz será la primera aula candidata a que se le asigne una asignatura.

Para terminar con la inicialización se dan por programadas las asignaturas del cuatrimestre no elegido y, a continuación, se estima la capacidad total disponible para de los diferentes días y turnos, sumando la capacidad disponible en todas las aulas que se encuentran libres para ese día y turno. Esto permitirá tener una idea de la capacidad disponible en cada uno de los periodos.

Una vez se ha inicializado el algoritmo, la asignatura seleccionada, que al ser la primera iteración se trata de la asignatura con el mayor número de matriculados, se propone como candidata para asignarse.

4.3.3.2. Construcción de la solución

Para construir la solución, véase reflejado en la figura 13, se programa un bucle que va recorriendo las asignaturas impartidas en la ETSII y que previamente han sido ordenadas según el número de estudiantes matriculados. Si la asignatura ha sido marcada como asignada, esta no se tiene en cuenta, y se pasa a la siguiente asignatura del listado. Cuando una asignatura aún no ha sido asignada, significa que se debe programar un examen para ella, por lo que, en primer lugar, se comprueba el turno en el que se debería programar designando si es de 1º, 3º o 5º un turno (de mañana si es el primer cuatrimestre o el de tarde si es el segundo) y las asignaturas de 2º y 4º curso, al contrario. A la vez que se realiza esta acción, se comprueba el listado de asignaturas optativas, en caso de estar en el listado se le asigna el turno contrario al correspondiente a su curso; en caso contrario, se recoge el turno en que se debe programar la asignatura.

A continuación, se comprueba si la asignatura analizada es poseedora de equivalentes, lo que permite asignarlas como un único bloque. Si posee dicha característica, se agrupan todos sus equivalentes en una única asignatura a los ojos de la asignación, formada por los diferentes códigos de estas, pero con un número de matriculados común ya que se van a asignar en el mismo slot de tiempo. Al agruparse las asignaturas consideradas equivalentes, estas ya se considerarían asignadas por lo que ya no se van a tener en cuenta en consecuentes iteraciones ya que se van a asignar a las mismas aulas que sus símiles.

En este punto, se va a seleccionar el día en el que se va a realizar el examen de la asignatura candidata. Aquí, se comprueba la compatibilidad de la asignatura con el día, en cuyo caso se pasa al día siguiente. Además, ese día debe de cumplir la condición de que su capacidad total disponible debe ser superior al número de los matriculados de la(s) asignatura(s).

Una vez seleccionado el día en que se va a programar el examen de una asignatura, y sus equivalentes en caso de tener, se recorre con un bucle el listado de aulas ordenado de mayor a menor capacidad. Se localiza la primera aula que se encuentra disponible para ese día, y se comprueba si el aula se encuentra disponible en el día seleccionado (información que se halla en la matriz disponibilidad).

Acto seguido, se compara el número de matriculados de la asignatura y equivalentes con la capacidad del aula seleccionada. Aquí encontramos dos vertientes, el número de matriculados es menor que la capacidad del aula con mayor capacidad o el número de matriculados es mayor que la capacidad del aula.

El número de matriculados es menor que la capacidad del aula con mayor capacidad

Si el número de matriculados es menor que la disponibilidad del aula con mayor capacidad designada para albergar el examen, se implementa una regla de decisión basada en el mayor ajuste posible del número de matriculados a la capacidad total del aula. Esto quiere decir que, para no desperdiciar aulas demasiado grandes que podrían albergar asignaturas con un gran número de matriculados, se analizan todas las aulas disponibles en el turno y día seleccionados con el objetivo de identificar si alguna de ellas se ajusta mejor al número de matriculados de la asignatura, disminuyendo la holgura capacidad - matriculados.

Para ello, antes de asignar la asignatura al aula, se comprueba si existe disponibilidad en otra aula cuya diferencia entre su capacidad y el número de matriculados sea mínima, es decir, que se reduce al máximo posible la holgura entre estas dos variables. Para ello, se entra en un bucle en el que se comprueba para todas las demás aulas en ese día si existe un aula que ajusta más su holgura que el aula seleccionada, si es así, esta pasa a ser el aula seleccionada hasta que se termina el bucle que recorre todas. Una vez hecho esto, si la asignatura posee equivalentes, se dan por asignadas todas ellas ya que realizarán el examen en la misma aula. Así, se actualiza la matriz `Horario_Solucion`, incluyendo la(s) asignatura(s) empleada(s) en la casilla correspondiente.

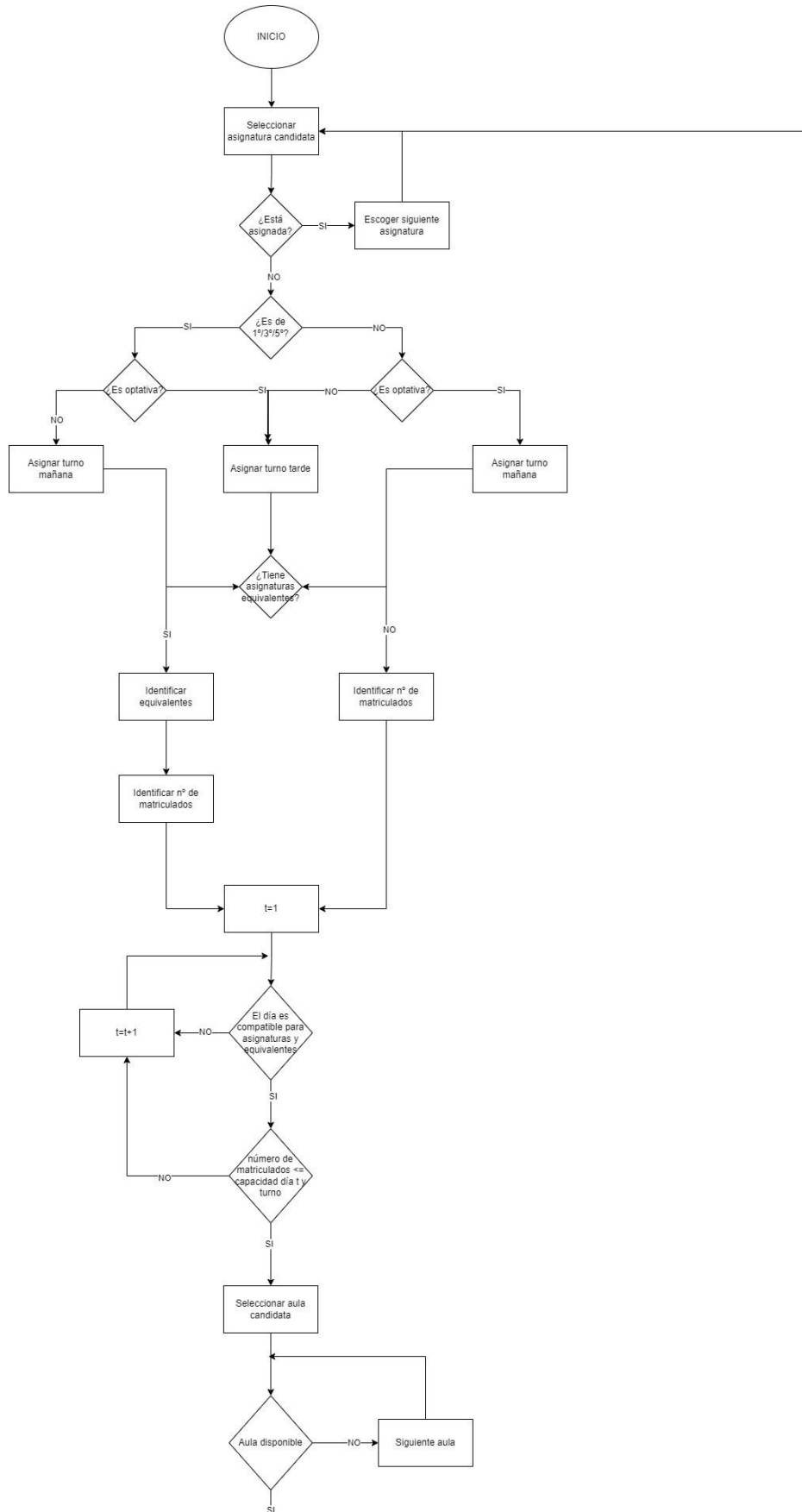


Figura 13. Diagrama de flujo de Construcción de la solución (Fuente: Elaboración propia)

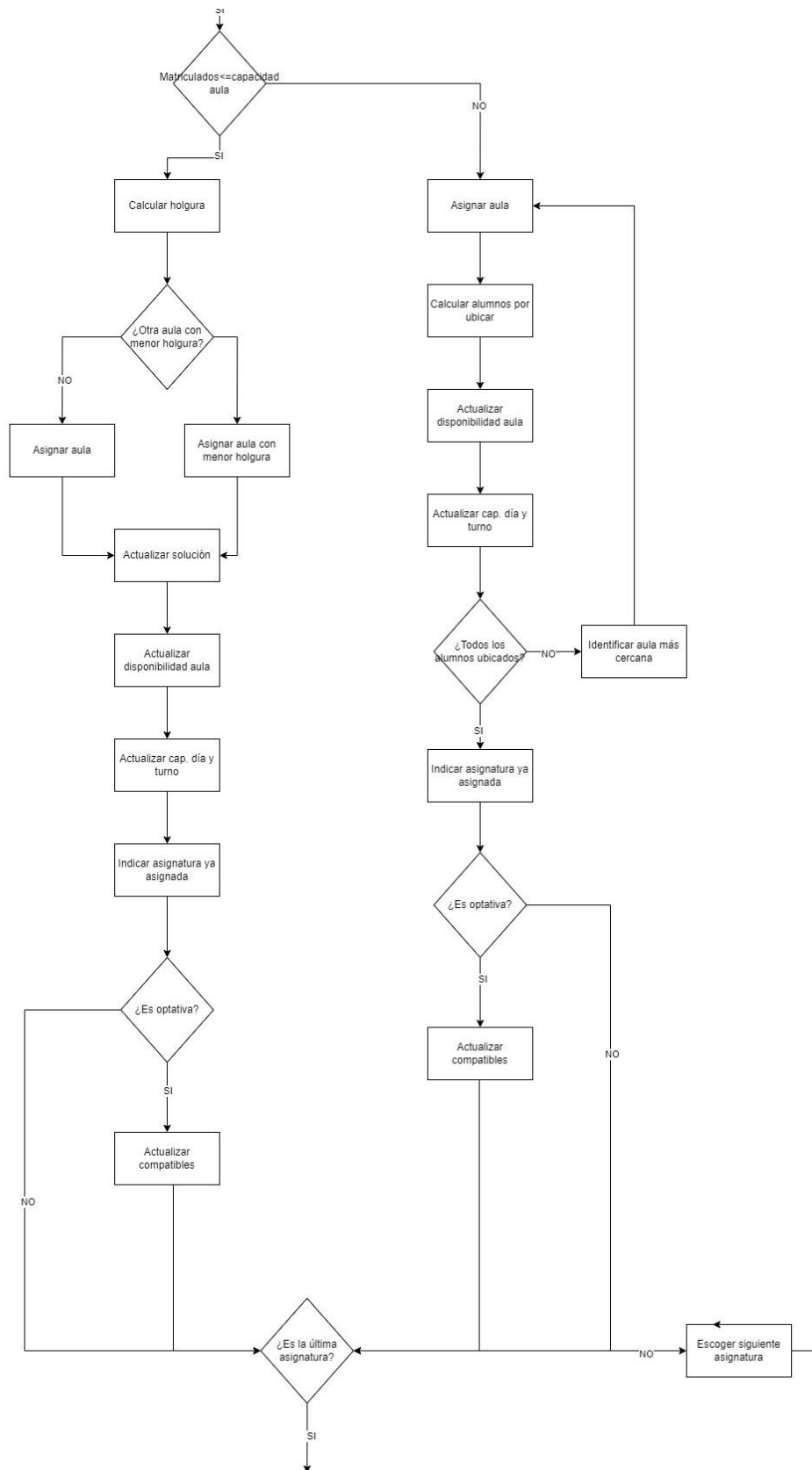


Figura 13. Diagrama de flujo de Construcción de la solución (Fuente: Elaboración propia)

El número de matriculados es mayor que la capacidad del aula

Si el número de los matriculados es superior al de la capacidad del aula designada y, por tanto, se han de utilizar más de un aula, se aplica la regla que sostiene que los matriculados restantes se asignarán en la clase más cercana. Si esto no fuera suficiente, se realizarían las iteraciones necesarias hasta que todos los alumnos matriculados en dicha asignatura se encuentren designados en las aulas correspondientes el mismo día.

Para ello, se asigna esa aula y se entra en un bucle que comprueba todas las asignaturas para ver cuál de ellas es la más próxima a la ya asignada se guarda el nuevo valor mínimo de distancia y el aula. Si no quedan alumnos por asignar, la iteración termina y se comienza la asignación de la siguiente asignatura. Si quedan alumnos por asignar se vuelve a buscar el aula cercana y así sucesivamente hasta que termina la asignación. Así, se actualiza la matriz Horario_Solucion, incluyendo la(s) asignatura(s) empleada(s) en la casilla correspondiente.

En definitiva, la función greedy que se pretende optimizar localmente es la distancia entre las clases asignadas, minimizando así este parámetro. A su vez, también intenta la minimización de la holgura capacidad-número de matriculados. Debido a esto es por lo que la regla fundamental de esta heurística es la de asignar a la clase más cercana disponible ya que, en su mayoría, los alumnos matriculados no suelen ajustarse bien al número del aula más grande de la que dispone la ETSII.

4.3.3.3. Finalización

Tras haber asignado ya la(s) aula(s) a la(s) asignatura(s) se actualiza la compatibilidad con respecto a su curso y turno y, a su vez, se actualiza la capacidad total de las aulas en el día utilizado, restándole así el número de matriculados que se han emplazado en esa jornada.

Antes de terminar, se imprime la solución guardada en la matriz "Horario_Solucion". Por último, se evalúa la solución, para ello, se calcula y acumula la distancia entra aulas empleadas para albergar el examen de una misma asignatura y la holgura total de las aulas utilizadas.

La heurística se da por concluida en el momento en el que todas las asignaturas de la Escuela tienen un slot de tiempo y clases asignadas en el periodo de exámenes. Consideraremos la solución factible si ocurre esto y no se produce ningún solape y todos los alumnos matriculados poseen una clase designada.

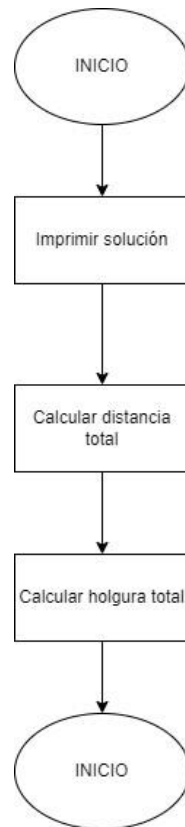


Figura 14. Diagrama de flujo de Finalización (Fuente: Elaboración propia)

4.4. Arquitectura de la herramienta heurística constructiva para la programación de exámenes ordinarios

Una vez que se ha creado un modelo heurístico para resolver el problema mencionado, debe implementarse en un lenguaje que el ordenador pueda comprender para que pueda usar un software comercial para obtener la solución ideal. La arquitectura de la herramienta propuesta para el caso de estudio se compone de los siguientes elementos técnicos, representados en la Figura 15:

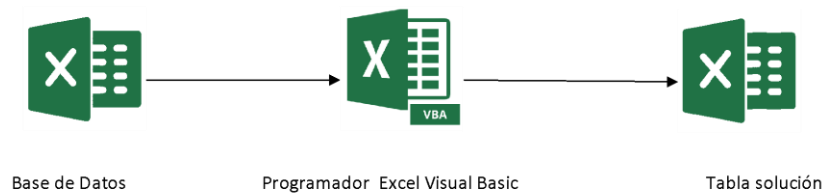


Figura 15. Arquitectura Técnica de la herramienta heurística para la programación de exámenes (Fuente: Elaboración propia)

1. MS Excel. Mediante el conocido programa, se crea la base de datos que permite trabajar al programa como punto de partida, y sirve, a su vez, como interfaz con el usuario en la que este puede indicar el cuatrimestre y turno a programar. Al final, solo con pulsar un botón, genera la heurística y la solución.
2. Programador Excel Visual Basic. Mediante el paquete de MS Excel que permite la programación en este, se plasma la heurística constructiva en código para que se pueda ejecutar las ordenes de la herramienta.
3. MS Excel. Como consecuencia de la programación de Visual Basic se genera el horario solución que presenta la solución óptima dada por la heurística de construcción.

En un primer plano, se genera en MS Excel la tabla de datos de las asignaturas, en ella se refleja el grado o postgrado al que se hace referencia, el cuatrimestre, el nombre de la asignatura, el curso, la vía, los respectivos códigos de las carreras y asignaturas y, por último, el número de matriculados. Por otro lado, encontramos tabuladas también las relaciones entre las asignaturas equivalentes, con sus correspondientes grado o postgrado al que se hace referencia, el cuatrimestre, el nombre de la asignatura, el curso, los respectivos códigos de las carreras y asignaturas y, por último, el número de matriculados. Además, se ha utilizado la tabla que refleja las aulas, relacionadas con su edificio y capacidad normal y en periodo de exámenes. Para concluir, podemos ver una última hoja que relaciona cada una de las clases de la escuela y expresa y en una tercera columna la distancia que hay entre ellas.

Por último, en el archivo de MS Excel se encuentra también el horario solución, en él se puede ver los días del periodo de exámenes dividido en dos turnos, mañana y tarde y, a su vez, subdividido en las diferentes clases que tienen cabida en la ETSII.

El código desarrollado para la implementación de la heurística constructiva se puede consultar en el Anexo I. Adicionalmente, se incluyen en los Anexos II, III y IV las funciones desarrolladas para identificar el tamaño de las matrices, realizar su lectura, y su ordenamiento, respectivamente.

4.5. Conclusiones

En este capítulo se ha presentado la propuesta de la herramienta heurística destinada a mejorar la programación de exámenes de una Escuela. A lo largo del capítulo se ha podido observar cómo se desarrolla la implementación de la propuesta y los elementos informáticos necesarios para que se pueda desarrollar eficientemente. Este análisis ha permitido no solo identificar las deficiencias del método actual, sino también proponer una solución innovadora que toma en cuenta múltiples factores para mejorar la eficiencia y funcionalidad del calendario de exámenes.

El objetivo principal de la herramienta heurística constructiva propuesta es reducir la distancia entre las aulas asignadas para los exámenes, lo que optimizará el uso de recursos y facilitará la logística del período de exámenes. Esta heurística se basa en un enfoque devorador, donde se prioriza la asignación de las asignaturas con mayor número de matriculados en las aulas de mayor capacidad disponibles y lo más cercanas posibles entre sí. El modelo creado se basa en una serie de restricciones duras y blandas que dirigen la asignación de exámenes.

Las restricciones rigurosas aseguran que no haya solapamientos de exámenes para un solo estudiante y que las asignaturas de un mismo curso se asignen en diferentes días y turnos. Las restricciones blandas, por otro lado, tienen como objetivo distribuir exámenes. Las restricciones blandas, por otro lado, tienen como objetivo asignar exámenes de un mismo grado con un día diferente y agrupar aulas en un mismo.

El algoritmo sugerido se implementa en múltiples pasos. Al principio, las asignaturas se ordenan según el número de matriculados y las aulas según su capacidad. Para reducir las distancias, las asignaturas más grandes se asignan a las aulas más grandes disponibles. Este método permite cambiar el cronograma de exámenes, modificando las tareas según la capacidad de las aulas y la distancia entre ellas, hasta completar la programación de todas las asignaturas sin problemas de solapamiento ni problemas de capacidad. La implementación de esta herramienta heurística constructiva proporciona varios beneficios importantes. En primer lugar, optimiza el uso del aula, reduce la distancia entre aulas y minimiza los recursos necesarios para monitorear y resolver incidencias durante los exámenes. En segundo lugar, mejora la experiencia de los estudiantes al reducir la duplicación y garantizar un diseño de examen más lógico y comprensible. Finalmente, proporciona una solución flexible y adaptable a futuros cambios en las estructuras de titulaciones, cursos y matrículas, facilitando la gestión administrativa y académica.

Microsoft Excel y Visual Basic for Applications (VBA) se utilizan para implementar técnicamente la heurística. Este método facilita la introducción y manipulación de datos y permite una interfaz amigable para el usuario. El programa creado en VBA maneja la base de datos de asignaturas, aulas y distancias y crea automáticamente el calendario de exámenes optimizado, considerando las restricciones y objetivos planteados.

En resumen, las herramientas heurísticas constructivas desarrolladas en este capítulo proporcionan una solución innovadora y eficaz para la planificación de pruebas en la ETSII. Al integrar múltiples factores y limitaciones en el proceso de asignación, esta herramienta no solo mejora la eficiencia operativa, sino que también enriquece la experiencia académica de los estudiantes y el personal. La metodología presentada se puede adaptar y extender a otras instituciones y situaciones y muestra potencial como una solución integral para la gestión del cronograma de auditorías.

5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS

5.1. Introducción

El Capítulo 5 presentará una herramienta heurística conocida como Algoritmo de Búsqueda Local, que se propone para solucionar los problemas de la programación de exámenes de un cuatrimestre en una escuela o facultad, especialmente para mejorar la programación de la ETSII de la UPV. Primero se establece la situación que presenta el problema a tratar. A continuación, el modelado, una herramienta heurística para la resolución de problemas. Finalmente, se presenta la resolución y se explica brevemente la arquitectura utilizada para implementar la heurística y su resolución.

Hipótesis

Las hipótesis implicadas en la realización del Algoritmo de Búsqueda total coinciden con las del capítulo anterior, ya que esta heurística se utiliza como complemento de la misma con la intención de poder mejorar la primera solución proporcionada.

5.2. Definición de la problemática

La problemática a resolver es basada en el capítulo anterior por lo que se toma como base todo lo mencionado anteriormente. Uno de los problemas de la programación de exámenes y el ETP es la gran complejidad que este problema entraña, es por esto que, mediante la heurística empleada en este capítulo se pretende tener en cuenta las soluciones y las posibilidades de mejora que se han podido “escapar” al utilizar la heurística constructiva.

5.3. Modelo de Algoritmo de Búsqueda Local para la programación de exámenes que minimiza la suma de las distancias de las aulas asignadas a la asignatura objeto

5.3.1. Base de datos y Restricciones

La base de datos utilizada para la realización de esta heurística sigue siendo la misma aplicada en el capítulo anterior, ya que es común. En esta base de datos recalamos que se encuentran todos los apuntes importantes en relación a tanto a las asignaturas como a las aulas y sus respectivas interrelaciones y codificación.

Adicionalmente ahora se cuenta con la matriz “Horario_Solucion” ya completa, en la que se encuentra recogida la solución propuesta por la heurística constructiva y sobre la que se partirá para realizar la búsqueda local, la matriz. Además, se encuentra la matriz “Compatibilidad”, que muestra la relación entre los grados, cursos y vías de las diferentes asignaturas y que serán de gran ayuda para saber hacia qué día se pueden permutar las asignaturas. En adición, se tiene también la matriz disponibilidad que alberga las aulas que se han quedado disponibles en los diferentes días.

Por último, cabe destacar que las restricciones tanto duras como blandas citadas en el anterior capítulo, son a su vez aplicables en el actual

5.3.2. Algoritmo de Búsqueda Local

5.3.2.1. Solución inicial

Para comenzar, cualquier heurística del tipo Búsqueda de entornos necesitan de una solución inicial de la que partir para poder comenzar la búsqueda en el espacio de soluciones al que se le llama S para poder realizar permutaciones que mejoren la solución tomada como base, todo esto como se ha desarrollado en el Capítulo 3. ANTECENTES.

La solución que se ha tomado como “solución inicial” es la obtenida a través de la heurística constructiva realizada en el Capítulo 4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS, la cual se supone como óptima ya que tiene en cuenta la minimización de la distancia entre las aulas asignadas a una asignatura. Como se ha observado anteriormente, esta heurística de tipo constructiva mediante la ordenación de mayor a menor número de alumnos tanto de las asignaturas (en función del número de matriculados) y las aulas disponibles (en función de la capacidad a la hora de la realización de una prueba de examen) pretende asignar las asignaturas que encabezan esta lista a las aulas con mayor capacidad que se encuentren habilitadas en ese momento. Además, para fomentar esto, si el número de matriculados es mayor que la capacidad del aula disponible elige entre las aulas más cercanas a esta y las asigna a su vez; y se tiene en cuenta la holgura entre la clase designada y la asignatura del examen.

La razón por la que se aplica esta heurística de búsqueda local es para intentar mejorar dicha asignación para comprobar si mejora la suma de las distancias de la asignatura elegida y, además, la holgura entre la capacidad del aula y el número de alumnos matriculados designados a la misma.

5.3.2.2. Generación de vecinos

Como se comentaba en relación a las heurísticas de búsqueda local, explora el espacio de soluciones S que contiene a todas las soluciones posibles de la problemática en cuestión (Carlos Andrés Romano, (2018)). Este espacio de soluciones contiene las soluciones que se conocen como vecinas, es decir, que parten de la misma base o solución inicial pero que debido a que se cambia una parte de la solución, son vecinas de esta.

En este caso, se seleccionará aleatoriamente la asignatura que se moverá en la matriz “Horario_Solución”. Para ello se selecciona aleatoriamente un día, turno y aula, lo que apunta a una casilla de la matriz “Horario_Solución”. En esta casilla puede haber almacenada una asignatura única, una asignatura con asignaturas equivalentes, o estar vacía (en caso en que el aula esté disponible en ese día y turno).

Al seleccionar una casilla con una o más asignaturas asignadas, se selecciona otro día aleatoriamente, denominado “día destino”, y se traslada esta asignatura o asignaturas al mismo. Para ello deberá de cumplir una serie de requisitos para ser asignada en el día de destino:

1. La capacidad disponible en el día destino debe de ser mayor o igual que el número de estudiantes matriculados de la asignatura/as candidata, si no es así, se deberá elegir otro día de forma aleatoria y volver a realizar las comprobaciones.
2. En el día destino no debe haber otras asignaturas programada que pertenezca a la misma titulación, curso y vía que la asignatura candidata. Si esto ocurriera se debería de escoger nuevamente de forma aleatoria y comenzar las comprobaciones de asignación. Existe una excepción a este requisito, cuando la asignatura es una asignatura optativa, dado que en este caso sí se puede programar más de una asignatura en el mismo día y turno.

A continuación, se comprueba si caben todos los alumnos matriculados en una única clase, si es así, intenta mejorar la holgura del número de matriculados con respecto de la solución de la heurística constructiva.

Si no es así, se procede a asignar las aulas como lo explicado anteriormente con la heurística constructiva, se elige la asignatura que se encuentra más cercana a la asignada y que esté disponible, así hasta que todos los alumnos tengan un aula designada.

Por último, se actualiza las aulas que se encuentran disponibles y las capacidades disponibles en los días escogidos. Para ello, se borra de la solución inicial la asignatura en todas las aulas que estaba previamente asignada, indicando que esas aulas quedan ahora disponibles, e incrementando la capacidad total del día y turno en el valor de la capacidad que tuvieran dichas aulas. Adicionalmente, en caso de no ser una asignatura optativa, se indicaría que ese día y turno pasa a ser compatible para la programación de otra asignatura de esa misma titulación, curso y vía. Por el contrario, en el día de destino, se asignan las nuevas aulas a la asignatura, indicando que estas ya no se encuentran disponibles, se decrementa la capacidad total de ese día y turno, y se indica que, en caso de no ser una asignatura optativa, ese día deja de estar disponible para la programación de exámenes de esa titulación, curso y vía.

5.3.2.3. Función de evaluación y criterio de mejora

La función elegida como función de evaluación o función objetivo y que mide la “calidad” de la solución obtenida es la que se obtiene a partir de la suma de las distancias que separan las aulas designadas para albergar la asignatura candidata para la permutación y la holgura de las aulas. En función de una fórmula sería:

$$FO = d_{ij} \cdot Y_{ijk}$$

Siendo d_{ij} la distancia que separa al aula i del aula j , e

Y_{ijk} una variable binaria que indica si el aula i y el aula j han sido asignadas para el examen de la asignatura k

$$H = \sum_i \sum_k (c_i - a_k) \cdot X_{ik}$$

Siendo c_i la capacidad del aula i , a_k el número de alumnos matriculados en la asignatura k y X_{ik} una variable binaria que indica si el aula i ha sido asignada a la asignatura k

De esta forma, se evaluará la distancia que separaba a las aulas asignadas a la asignatura candidata antes de realizar la permutación y después de realizar esta acción y la holgura de las aulas. Si la distancia obtenida tras realizar la permutación es menor que la que se tenía en la que se tenía en una primera instancia, se considera la nueva solución como “mejor” que la anterior y, en consecuencia, la sustituye como solución. Por otro lado, se tiene como contador la holgura para, tras finalizar el algoritmo, comparar con la solución inicial.

5.3.2.4. Condición de parada

El algoritmo explicado se ejecutará para 5, 10, 50, 100, 250, 500, 750 y 1000 iteraciones, y se analizará las soluciones obtenidas en estos experimentos en el próximo capítulo. Esto quiere decir que, tras realizar las acciones comentadas anteriormente y sustituir la nueva solución óptima, se vuelve al principio y se vuelve a comenzar con el Algoritmo de búsqueda local eligiendo nuevamente día, turno y aula de forma aleatoria. El algoritmo concluirá así cuando se llegue al número de iteraciones establecido.

5.3.2.5. Algoritmo paso a paso

El algoritmo comienza con la ejecución de la heurística constructiva, obteniendo así la solución de partida para el Algoritmo de Búsqueda Local, todo esto reflejado en la figura 14. Una vez ejecutada la primera heurística, el algoritmo objeto de este capítulo, comienza con la Inicialización de variables, donde se inicializan todas las variables necesarias, incluyendo las matrices y arreglos que se utilizarán en el algoritmo.

Tras haber inicializado las variables, comienza el Bucle Principal de Iteraciones, en el cual se establece un bucle para realizar un número fijo de iteraciones. El bucle comienza seleccionando de forma aleatoria un día aleatorio, un turno al azar y un aula al azar. A partir de esta generación se extrae la asignatura o asignaturas (dependiendo de si tiene equivalentes). A través de un vector binario, se distingue si esa aula se encuentra vacía o alberga una asignatura, ya que, si no, se pasaría a la siguiente iteración.

Una vez se tiene la asignatura a permutar, se registra mediante un bucle si existe más de una clase asignada a esa asignatura(s), ya que puede darse el caso de que se le haya asignado más de un aula. Ahora que se tiene tanto la asignatura objetivo como la clase o clases que la alberga se comprueba si es una única asignatura o si es un conjunto de asignaturas equivalentes.

En la siguiente fase se identifica el número de matriculados. Si se trata de una única asignatura, solo consiste en leer de la matriz "Datos_Asignaturas2, el número de matriculados asociados a la misma. Si se encuentra el caso en el que son un conjunto de asignaturas equivalentes, se lee a su vez sus capacidades de la matriz "Datos_Asignaturas" y se sumarían para conformar un único valor. Tras esto se comprueba si se trata de una asignatura optativa, para saber si se debe de asignar en un turno o en otro.

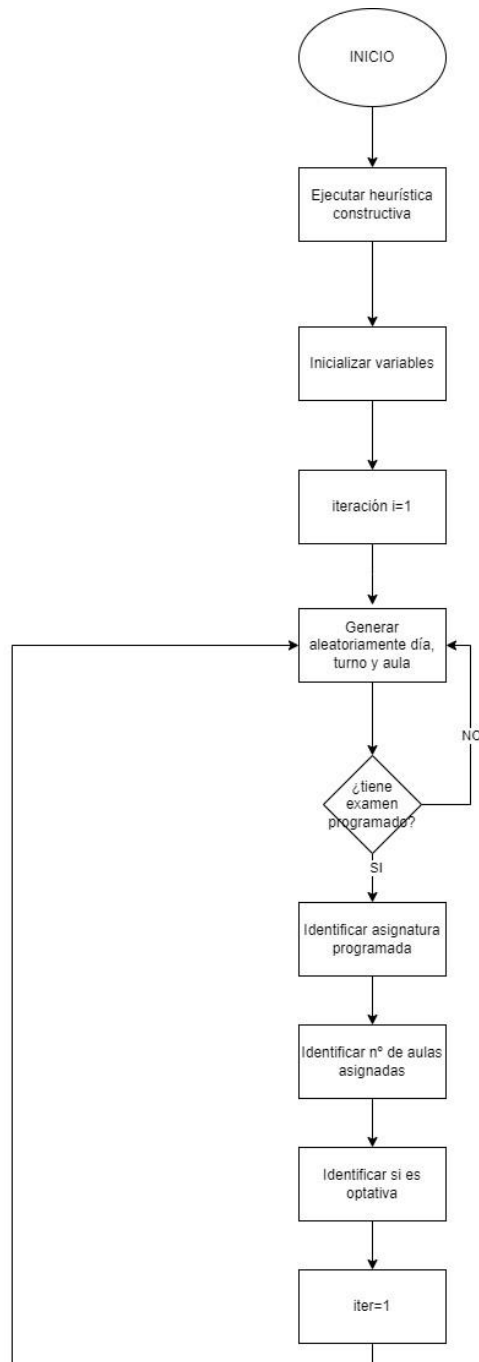


Figura 16. Diagrama de Flujo del Algoritmo Paso a Paso (Fuente: Elaboración propia)

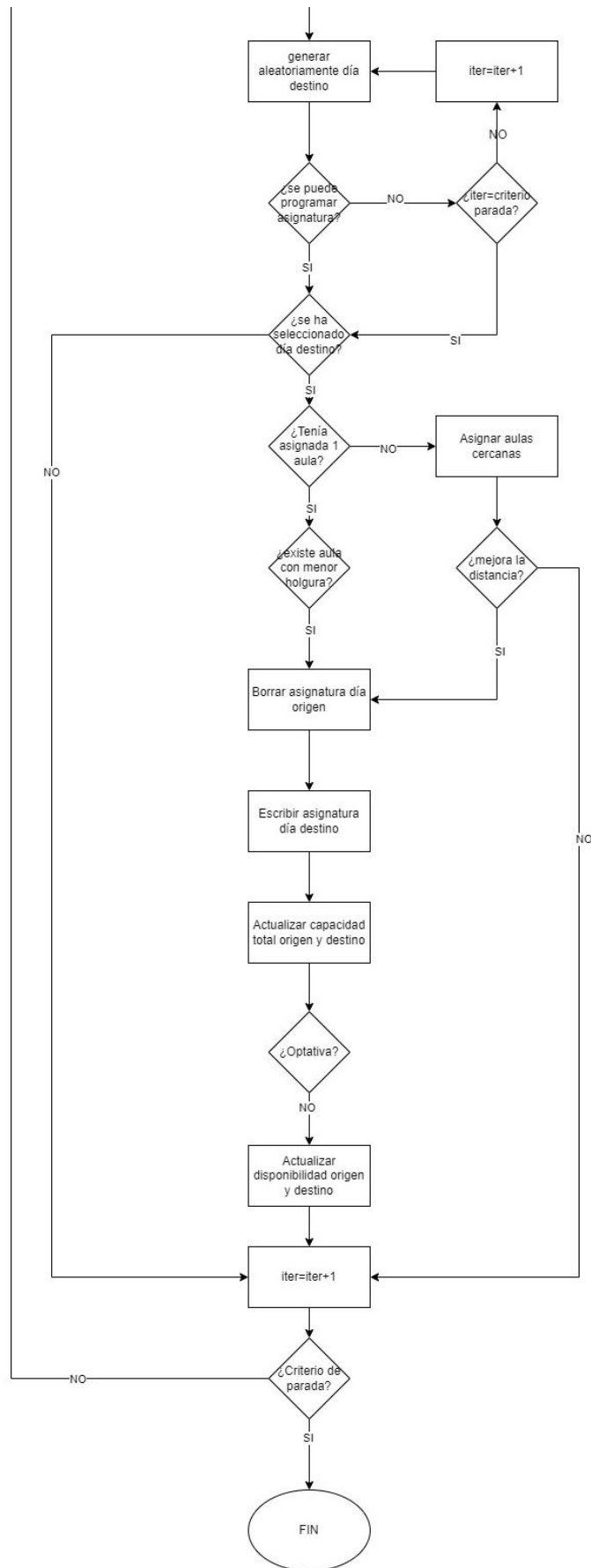


Figura 17. Diagrama de Flujo del Algoritmo Paso a Paso (Fuente: Elaboración propia)

Ahora, se realiza la elección del día de destino de forma aleatoria, tras haber identificado el día de destino, hay que ver si ya hay otra asignatura de ese curso y titulación en ese día (esto si no es optativa), además se vuelve a comprobar si son equivalentes para comprobar la compatibilidad de todas. Además, se comprueba también la capacidad total del día, para ver si puede albergar la realización de ese examen. Si se cumplen ambas condiciones, se asigna el día de destino.

Tras encontrar el día de destino, se busca el aula donde se realizará el examen, y se dan dos situaciones: inicialmente tenía un aula, o se encontraba repartido en diferentes aulas.

Si se encuentra en el primer caso, el caso en que la asignatura/asignaturas seleccionada/s solo tenga asignada un aula en inicialmente, se comprueba el aula que era y su holgura y comienza la asignación. Interesa que evalúe otra aula que esté más ajustada en capacidad ya que de esta manera liberaría aulas más grandes para poner otras asignaturas. Para ello se realiza un bucle que recorre todas las aulas de ese día y comprueba si están disponibles. Si es así, y la asignatura cabe en el aula, se calcula su holgura, y si y sólo si mejora se asigna esa aula. Así, se borra del día origen la asignatura objetivo y sus equivalentes si las tiene y se escriben en el día de destino. Por último, se actualiza la capacidad total de ambos días, la disponibilidad y el horario solución.

Si se da el caso de que inicialmente tenía más de un aula, se procede como en la heurística constructiva, pero teniendo en cuenta el día de destino y viendo las aulas que quedan allí disponibles. Se busca el aula en el día de destino y si puede albergar todos los matriculados se asigna el aula ya que así la distancia sería 0. En caso contrario, se asigna al aula analizada anteriormente y se localiza el aula más cercana posible. Si y sólo si las aulas de destino mejoran a las anteriores se borra del día origen la asignatura objetivo y sus equivalentes si las tiene y se escriben en el día de destino. Por último, se actualiza la capacidad total de ambos días, la disponibilidad y el horario solución.

Para concluir se imprime la solución y se evalúa la solución obtenida con respecto a lo que a la distancia total entre aulas asignadas a una asignatura y la holgura se refiere.

5.4. Arquitectura de la herramienta heurística de búsqueda local para la programación de exámenes

Tras haber realizado el Algoritmo de búsqueda local propuesto, se plantea la implementación en un lenguaje que el ordenador pueda comprender para que pueda usar un software comercial para obtener la solución ideal. La arquitectura de la herramienta propuesta para el caso de estudio se compone de los siguientes elementos técnicos, representados en la Figura 18:



Figura 18. Arquitectura Técnica de los Algoritmos de búsqueda local para la programación de exámenes (Fuente: Elaboración propia)

1. MS Excel. En una primera instancia, se parte de la Tabla Solución obtenida en el apartado anterior para la Heurística constructiva.
2. MS Excel. A su vez, se parte también de la base de datos realizada para la heurística del apartado anterior, ya que es compartida. Se recuerda que en ella se encuentran los datos relativos a las asignaturas y a las aulas y su disposición.
3. Programador Excel Visual Basic. Mediante el paquete de MS Excel que permite la programación en este, se plasman ambos Algoritmos de búsqueda local en código para que se pueda ejecutar las ordenes de la herramienta.
4. MS Excel. Como consecuencia de la programación de Visual Basic se genera el horario solución que presenta la solución óptima dada sendos algoritmos.

En un primer lugar, se aprovecha la tabla solución obtenida en MS Excel en el Capítulo 4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS, la cual servirá de solución inicial. A continuación, se aprovecha la ya generada en MS Excel tabla de datos de las asignaturas, en ella se refleja el grado o postgrado al que se hace referencia, el cuatrimestre, el nombre de la asignatura, el curso, los respectivos códigos de las carreras y asignaturas y, por último, el número de matriculados. Por otro lado, encontramos tabuladas también las relaciones entre las asignaturas equivalentes, con sus correspondientes grado o postgrado al que se hace referencia, el cuatrimestre, el nombre de la asignatura, el curso, los respectivos códigos de las carreras y asignaturas y, por último, el número de matriculados. Además, se ha utilizado la tabla que refleja las aulas, relacionadas con su edificio y capacidad normal y en periodo de exámenes. Para concluir, podemos ver una última hoja que relaciona cada una de las clases de la escuela y expresa y en una tercera columna la distancia que hay entre ellas.

Por último, en el archivo de MS Excel se encuentra también el horario solución, en él se puede ver los días del periodo de exámenes dividido en dos turnos, mañana y tarde y, a su vez, subdividido en las diferentes clases que tienen cabida en la ETSII.

El código desarrollado para la implementación de la heurística de búsqueda local se puede consultar en el Anexo V.

5.6. Conclusiones

El capítulo 5 aborda el uso de una herramienta heurística, específicamente un Algoritmo de Búsqueda Local, para optimizar la programación de exámenes en la Escuela Técnica Superior de Ingeniería Industrial de la Universidad Politécnica de Valencia. Se describe el problema a resolver, el modelo heurístico propuesto y la implementación de la solución.

La programación de exámenes debe gestionar múltiples grados y cursos, dividiéndose en dos cuatrimestres, se deben coordinar pruebas de medio trimestre, fin de trimestre y un examen general. La estructura del horario sigue reglas específicas, como asignar turnos de mañana y tarde según el curso, evitando solapamientos de exámenes para estudiantes con materias de diferentes años, además, se busca minimizar el uso de aulas y la distancia entre ellas para optimizar recursos y reducir costes.

La herramienta consta del algoritmo que busca minimizar la distancia total entre las aulas asignadas para cada examen. Utiliza una base de datos con información de asignaturas y aulas, y se aplican restricciones duras (reglas que no pueden violarse) y blandas (preferencias que deben cumplirse en la medida de lo posible). El algoritmo comienza con una solución inicial, se utiliza una heurística constructiva que minimiza la distancia entre aulas asignadas a una asignatura, a continuación, se exploran soluciones vecinas cambiando la asignación de aulas y turnos de manera aleatoria, asegurando cumplir con las restricciones. Se evalúa la función objetivo, la solución basada en la distancia total entre aulas y, si la nueva solución es mejor, se adopta como la nueva solución óptima. Por último, se detallan las condiciones de parada, el algoritmo se ejecuta por un número fijo de iteraciones (5, 10, 50, 100, 250, 500, 750 y 1000) y se analiza el resultado en cada caso.

La implementación se realiza en MS Excel con programación en Visual Basic. La arquitectura incluye:

1. Tabla de solución inicial y base de datos en Excel.
2. Código en Visual Basic para ejecutar los algoritmos.
3. Generación del horario solución optimizado.

La herramienta usa las tablas de datos de asignaturas, relaciones entre asignaturas equivalentes, aulas y distancias entre ellas. El resultado es un horario detallado con asignaciones de exámenes optimizadas según las restricciones y criterios establecidos.

En definitiva, este capítulo describe un enfoque heurístico para mejorar la programación de exámenes, buscando optimizar el uso de recursos y minimizar la distancia entre aulas. Se presenta un modelo de búsqueda local y se detalla la implementación técnica, ofreciendo una solución práctica y eficiente para la planificación académica.

6. APLICACIÓN A LA ETSII. ANÁLISIS DE RESULTADOS

6.1. Introducción

Una vez abordadas las dos diferentes herramientas heurísticas, con su posterior explicación estructural de los diferentes modelos de forma ejecutable, se procede a la puesta en escena de las dos heurísticas con el caso aplicado del ETSI de la Universidad Politécnica de Valencia, con sus asignaturas y suponiendo un periodo de exámenes del cuatrimestre A, para ambos turnos, a lo largo de 10 días.

En una primera instancia, actualmente la programación de exámenes de la que se plantea el caso real, la ETSII, se asigna de forma manual, partiendo del horario del año pasado y realizando una permutación de un día al día siguiente. Esto quiere decir que si, por ejemplo, el tercer día del periodo de exámenes se encuentran asignadas las asignaturas 11450 y 11449, las citadas se permutan al cuarto día del periodo de exámenes, y las del cuarto día al quinto, así sucesivamente. El calendario se exporta a través de la aplicación ALGAR, que luego alimentará otras aplicaciones de la ETSII que permiten su visualización gráfica una vez que se complete.

A día de hoy, la problemática la creación de los horarios y calendarios de la época de exámenes de la ETSII es considerada de una índole compleja, un problema complejo de asignación de los exámenes a las fechas y aulas. Además, en relación al proceso de asignación actual, uno de sus principales inconvenientes radica en que no tiene en cuenta ningún parámetro de medición de su optimalidad o de detectar incidencias y que, al final y al cabo, se arrastran de un año en otro. Es por esto que se propone intentar minimizar la distancia entre las aulas, favoreciendo así que se necesiten menos recursos a la hora de la realización de esta clase de pruebas.

6.2. Solución de la Heurística Constructiva

En primer lugar, encontramos la solución propuesta para la resolución de la problemática de la programación de exámenes de la ETSII mediante la heurística constructiva, que pretende minimizar la distancia entre las aulas asignadas a una misma asignatura y la holgura en el caso de la asignación a una única aula.

Como es observable en la Figura 19 (asignación del primer cuatrimestre en el turno de mañana) y en la Figura 20 (asignación del primer cuatrimestre en el turno de tarde), la asignación que se realiza de las asignaturas es, en su mayoría, en aulas muy próximas o que, al menos, se encuentre en el mismo edificio.

Por ejemplo, la asignatura 33809 “Calor, Frío y Climatización” del MU11, cuenta con un total de 383 matriculados y ha sido asignada en las aulas 011, 012, 013, 014, 021, 022 y 023, que cuentan con una capacidad total de 404 estudiantes y que se encuentran todas ellas en las dos primeras plantas del aulario.

Para el caso de las asignaturas equivalentes, se observa como estas se programan en un mismo día y turno, como es el caso de las asignaturas equivalentes 11426 “Estructuras” de GITI y 11529, “Estructuras” de la rama 4B de GIOI que se asigna a las aulas 031, 032, 033, 034, 035 y 041.

Por otra parte, hay que tener en cuenta que las aulas de la familia 400 y 500 se encuentran en edificios conectados por lo que las distancias entre sus aulas son muy pequeñas, pudiéndose asimilar como un único edificio. A su vez, se puede observar que no se solapan exámenes de asignaturas correspondientes a los mismos los cursos ni vías para las asignaturas de la misma titulación, viéndose así que se cumplen todas las hipótesis planteadas. Un ejemplo de esto sería la asignatura 11470 “Matemáticas I” de GIOI programada para el día 1, y que no presenta solapamiento con ninguna asignatura de su curso (1º) y mismo cuatrimestre, como son: Expresión Gráfica (11475), Física I (11472) y Química (11473).

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 023	33809	33804	11402	33805	33813	11404	34079	11473	11490	10169
	AULA 024				34368			34369	34370		
	AULA 025										12950
	AULA 031	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499	12262	12934	11472	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499	12262	12934	11472	11407	11426;11529	11416	11473	11490	10169
	AULA 110						34358				
	AULA 111	5386;33786;3474	14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112										
	AULA 211										
	AULA 212				14661	14666			13036	12285	34382
	AULA 213										
	AULA 214	13444	30369	32603	30455		32604				
	AULA 215	33905	34373	32802	56126	33900	34371	34377	33903	33901	33898
	AULA 222					11478					
	AULA 223										
	AULA 311	34935	34934	33906			14240	34555	34931		
	AULA 312	12265	13034	11494	12268	13040	13963	11497	34381	14004	12297
	AULA 323	12265	13034	11494	12268	13040	13963	11497	13036	12285	12297
	AULA 410	13755	13884	34083	34082	34938			34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663	56372	56127		56128	
	AULA 420	32804	56392	56394	32803	31831	56391	32801	56393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12956	13060	13054	13053	13066
	AULA 424	13045	13058	13917	11498	12287	12956	13060	13054	13053	13066
	AULA 425										
	AULA 521	12275	12274	12291	12941	12946	33465	12951	12937	12936	33466
	AULA 522	12275	12274	12291	12941	12946	33465	12951	12937	12936	33466
	AULA 523	34367	33473	33464	33468						
	AULA 524				14020	30381	14030	14022			32606
	AULA 525	34367	33473	33464	33468						
	AULA 526										
	AULA 527										

TURNO MAÑANA

Figura 19. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)

TURNNO TARDE												
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 022	33749											
AULA 023												
AULA 024							33706		33476			
AULA 025												
AULA 031	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508		
AULA 032	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508		
AULA 033	11421;11524	11506	11505	12954	12949	12266	12963	11489	12942	12960		
AULA 034	11421;11524								33707			
AULA 035	11421;11524											
AULA 041	33749	11506	11505	12954	12949	12266	12963	11489	12942	12960		
AULA 110												
AULA 111	33764	33717	33683	33682	31823	33732;33741	33763	33478	33485	33695		
AULA 112												
AULA 211												
AULA 212	33684				33718		33488	33752	33719	33714		
AULA 213	33480				33681		33757	33723	33487	33733		
AULA 214												
AULA 215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481		
AULA 222												
AULA 223												
AULA 311												
AULA 312	12961	12272	33479	13032	33703			33702	33736	33704		
AULA 323	12961	12272	12273	13032	12295	33486	31927	13031	13029	33744		
AULA 410					12295	13059	12280	13031	13029	33744		
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688		
AULA 412	33727			33724	33756	33728	33737	33739		33740		
AULA 420	33729	33491	12273	33730	33483	13059	12280	33489	34762	31826		
AULA 421	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948		
AULA 424	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948		
AULA 425												
AULA 521		13062	13033	13068	12288	12298	13051	12957		12294		
AULA 522		13062	13033	13068	12288	12298	13051	12957		12294		
AULA 523		13041	12947	33705;35387		12939						
AULA 524	11487	33678	33722	33753	33684	33755	33721	33720	33761	33760		
AULA 525		13041	12947	33705;35387		12939						
AULA 526												
AULA 527	33738								33679			

Figura 20. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

6.3. Solución del Algoritmo de Búsqueda local

Manteniendo el caso en el que se programa el primer cuatrimestre para ambos turnos, se presenta la solución del Algoritmo de Búsqueda Local, realizando 500 iteraciones, para que se tuviera un número considerable de las mismas.

Como se puede ver en la Figura 21 (programación del primer cuatrimestre en el turno de mañana) y la Figura 22 (programación del primer cuatrimestre en el turno de tarde), se han realizado un número total de 115 permutaciones de asignaturas tras las 500 iteraciones. Esto quiere decir que, en 385 iteraciones, el algoritmo no ha conseguido mejorar la distancia entre las aulas asignadas a la asignatura candidatas. En todas las permutaciones realizadas se cumplen con todas las restricciones e hipótesis incluidas, cumpliendo con la compatibilidad de los cursos y turnos.

A continuación, se muestran tres ejemplos que comprueban lo sostenido:

Asignatura 11472 Física I.

En la programación obtenida con la heurística constructiva, esta asignatura del grado de GIOI se encontraba asignada a las aulas 033 y 041, cuya distancia es de 74 pasos, mientras que en la nueva solución propuesta que emplaza la asignatura en las aulas 311 y 323, que se encuentran separadas por 61. Como es observable, esto supone una reducción de 13 pasos en su distancia, lo que supone una mejora del 17,56%.

Asignatura 12280 Proyectos

Proyectos de Ingeniería Química. Si se presta atención en la heurística constructiva, esta asignatura de GIQ se encuentra asignada en las aulas 323 y 420, que se encuentran separadas por una distancia de 322 pasos, mientras que en la solución designada por el Algoritmo de Búsqueda Local encontramos la realización del examen en las aulas 312 y 323, separadas por 61 pasos. Esto se traduce en una mejora del 18,9%.

Asignatura 12939 Ciencia de Materiales.

La heurística constructiva coloca la asignatura en cuestión del grado de GIE en las aulas 523 y 525 separadas por 5 pasos y las traslada al aula 323, no teniendo ninguna distancia ya que se emplaza una única aula.

Como se puede observar en los ejemplos propuestos además de cambiar de aula, también se cambia de día, todo esto sin alterar las restricciones iniciales y se produce un ahorro de distancia total entre aulas de 2.431 pasos, pasando de una distancia total de 17.458 pasos (con la heurística constructiva) a una de 15.027 pasos, suponiendo un ahorro del 13,93%.

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 012	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 013	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 014	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 021	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 032	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 023	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 034	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 025					34368			11473		
	AULA 031	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	12960
	AULA 032	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499				11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499		13066		11407	11426;11529	11416	11473		34367
	AULA 110						34358	12268			
	AULA 111	35386;33786;34766	14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112							12268			
	AULA 211										
	AULA 212				14661						34382
	AULA 213										
	AULA 214	13444	30369	32603	30455		32604				
	AULA 215	33906	34373	32802	35126	33900	34371	34377	33903	33901	33898
	AULA 222				12297	11478	12285	13040			
	AULA 223				12297		12285	13040			
	AULA 311	34935	34934	33906		11472	14240	34555	34931		
	AULA 312			11494			13963		34381	14004	
	AULA 323	33473		11494		11472	13963				
	AULA 410	13755	13884	34083	34082	34938			34084	34932	34933
	AULA 411	14011	32807	34372	14662;14666	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663	35372	35127	14666	35128	
	AULA 420	32804	35392	35394	32803	31831	35391	32801	35393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12356	13050	13054	13053	12362
	AULA 424	13045	13058	13917	11498	12287	12356	13050	13054	13053	12362
	AULA 425				13034		11497				
	AULA 521	12275	12274	12291	12941	12946	33465	12361	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	33466	12361	12937	33470	33466
	AULA 523		11490	33464	12934	12934		10169	12265		13036
	AULA 524				14030	30381	14030	14022			32606
	AULA 525		11490	33464	13034	12934		10169	12265		13036
	AULA 526										
	AULA 527						11497				

Figura 21. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)

TURNO TARDE													
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 022													
AULA 023													
AULA 024							33706		33476				
AULA 025													
AULA 031	11421;11524	11492	11477	11485	13029	11486	11507	12273	12293				
AULA 032	11421;11524	11492	11477	11485	13029	11486	11507	12273	12293				
AULA 033	11421;11524	11506		1254				11489	12942				
AULA 034	11421;11524								33707				
AULA 035	11421;11524												
AULA 041		11506		1254		12282		11489	12942				
AULA 110													
AULA 111	33764	33717	33683	33682	31823	33732;33741	33763	33478		33695			
AULA 112													
AULA 211													
AULA 212	33684				33718		33488	33752	33719	33714			
AULA 213	33480				33681		33757	33723	33487	33733			
AULA 214													
AULA 215	33689	34761	33701		33686	31825	33691	33690	33687	33481			
AULA 222	13069												
AULA 223	13069												
AULA 311					33703			33702	33736	33704			
AULA 312	12961		12960		12256	33486	31827	12380	12380	33744			
AULA 323	12961		12960		12256			12339	12280	33744			
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688			
AULA 412	33727			33724	33756	33728	33737	33739	33755	33740			
AULA 420	33729	33491	33762	33730	33483	33760	33485	33489	34762	31826			
AULA 421	12272	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 424	12272	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 425	12949	12963				13031			33749				
AULA 521	11508	13062	13033	13068	12288	12298	13051	12957	11481	12294			
AULA 522	11508	13062	13033	13068	12288	12298	13051	12957	11481	12294			
AULA 523		13041	12947	33705;33738	11488		12266		11506				
AULA 524	11487	33678	33722	33753	33694		33721	33720	33761				
AULA 525		13041	12947	33705;33738	11488		12266		11506				
AULA 526	12949	12963				13031			33749				
AULA 527	33738								33679				

Figura 22. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

6.4. Comparación de resultados

En este apartado se va a comparar la eficiencia de la heurística constructiva con la del Algoritmo de Búsqueda Local al realizar 5, 10, 50, 100, 250, 500, 750 y 1000 iteraciones basándose en 5 indicadores que se han propuesto como punto de referencia (todos esos resultados se encuentran en el ANEXO):

- Distancia total
- Holgura total
- Número de aulas empleadas
- Número de asignaturas reprogramadas
- Tiempo de resolución

Los resultados obtenidos para estos indicadores se muestran en la Tabla 1, que se presenta en la página siguiente. En este análisis es observable que, aunque la heurística constructiva propuesta es muy sólida, al aplicar la heurística de búsqueda local se produce una mejora considerable con respecto a la distancia total existente entre aulas asignadas a una misma aula, que se disminuye conforme aumenta el número de iteraciones (Figura 23).

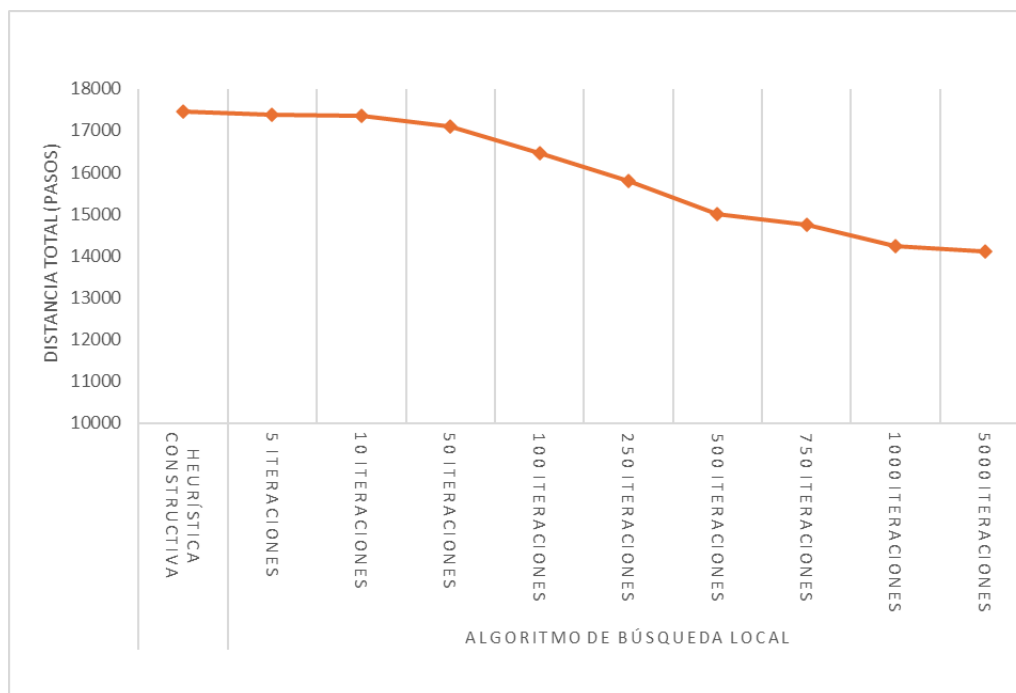


Figura 23. Gráfica comparativa de la eficiencia de las heurísticas (Fuente: Elaboración propia)

Por otro lado, conforme aumenta el número de iteraciones es observable también que aumenta ligeramente la holgura ya que esta solo se mejora obligatoriamente si la asignatura sobre la que se ejerce la permutación tiene asignada una única aula. Esto indica que, realizar el cambio de las aulas asignadas a una asignatura para reducir su distancia a veces implica emplear aulas más grandes, estando de esta manera más lejanas a un ajuste entre la capacidad requerida y la asignada. Además, parece que la minimización de la distancia total y la minimización de la holgura son dos objetivos que están en conflicto, puesto que mejorar el valor de uno de ellos, implica en la mayoría de las ocasiones empeorar el valor obtenido por el otro.

Además, teniendo su punto de inflexión en las 1000 iteraciones el número de aulas empleadas para la programación de exámenes se reduce levemente. Sin embargo, el número de aulas empleadas permanece bastante estable a lo largo de toda la experimentación realizada.

Por otro lado, se puede observar como el número de asignaturas reprogramadas crece de forma asombrosa conforme se añaden las iteraciones, no habiendo mucha diferencia entre iteraciones de grado próximo, pero destacado entre 5 iter., 500 iter. Y 5000 iter. Esta relación se puede ver representada de forma racional (número de asignaturas reprogramadas/número de iteraciones), siendo un 0% en el primer caso, 0,288 en el segundo y de 0,0418 en el tercer caso, luego se puede ver cómo encuentra la optimalidad en las 500 iteraciones.

Por último, cabe destacar con respecto a lo que a la eficiencia computacional se refiere que el tiempo de respuesta del algoritmo es bastante pequeño y parecido, aunque se dispara en el caso de las 500 iteraciones (lo cual es coherente si tenemos en cuenta el número de permutaciones que realiza) aunque sigue siendo bastante rápido. El rápido tiempo de resolución ofrecido por la herramienta desarrollada la convierte en una herramienta robusta y de gran utilidad para su aplicación en el entorno real de la ETSII.

		INDICADOR				
		Distancia total (pasos)	Holgura total (alumnos)	Número de aulas empleadas	Nº de asignaturas reprogramadas	Tiempo de resolución
HEURÍSTICA CONSTRUCTIVA		17458	4909	504	-	0,47"
5 iteraciones		17389	4893	504	4	1,38"
10 iteraciones		17361	4927	504	12	1,46"
50 iteraciones		17096	4894	505	25	1,56"
100 iteraciones		16468	5401	503	38	1,54"
250 iteraciones		15796	5621	502	92	1,53"
500 iteraciones		15008	6600	503	144	2,06"
750 iteraciones		14752	6291	497	164	2,45"
1000 iteraciones		14238	7488	495	160	2,78"
5000 iteraciones		14105	9869	496	209	7,65"
ALGORITMO DE BÚSQUEDA LOCAL						

Tabla 1. Comparación de eficiencia de las heurística (Fuente: Elaboración propia

6.5. Conclusiones

En este capítulo se han evaluado los dos enfoques heurísticos distintos para la programación de exámenes en la ETSII de la Universidad Politécnica de Valencia, con el objetivo de mejorar la asignación de aulas y reducir la distancia entre ellas durante el periodo de exámenes.

Primero, se ha abordado la solución a través de la heurística constructiva. Este método demostró ser efectivo al asignar las aulas de manera que se minimizara la distancia entre ellas, asegurando que la mayoría de las asignaturas se ubicaran en aulas cercanas o dentro del mismo edificio. Ejemplos como la asignación de "Calor, Frío y Climatización" y "Estructuras" muestran cómo esta heurística cumple con las restricciones planteadas y mejorar la distribución espacial de las aulas, optimizando la logística y la utilización de recursos.

Posteriormente, se ha presentado el Algoritmo de Búsqueda Local, que, tras 500 iteraciones, ofreció mejoras significativas en términos de distancia total recorrida entre aulas. Los ejemplos de asignaturas como "Física I", "Proyectos de Ingeniería Química" y "Ciencia de Materiales" ilustran cómo el algoritmo puede reducir sustancialmente las distancias, proporcionando un ahorro notable en el desplazamiento entre aulas y, por ende, en los recursos necesarios para la gestión de exámenes.

La comparación entre ambos métodos a través de múltiples iteraciones y diversos indicadores muestra que, aunque la heurística constructiva es sólida y eficiente, el Algoritmo de Búsqueda Local logra mejoras adicionales conforme aumenta el número de iteraciones. A pesar de un incremento en la holgura total y el tiempo de resolución computacional el beneficio en la reducción de distancia y la optimización del uso de aulas es evidente. En definitiva, la aplicación de estos métodos heurísticos proporciona soluciones viables y mejoradas para la compleja tarea de programación de exámenes en la ETSII. Mientras que la heurística constructiva ofrece una base robusta, el Algoritmo de Búsqueda Local optimiza aún más los resultados, demostrando que un enfoque iterativo puede generar beneficios considerables en la organización y eficiencia de los horarios de exámenes. Esto sugiere que una combinación de ambas heurísticas podría ser la estrategia óptima para futuras implementaciones, aprovechando las fortalezas de cada método para lograr una programación de exámenes aún más efectiva y eficiente.

Por otro lado, mediante la utilización de estas herramientas, se tiene como objetivo la contribución a las metas relacionadas con el desarrollo sostenible, es decir, con los Objetivos de Desarrollo Sostenible (ODS) propuestos por la ONU (ONU, 2024).

En concreto, mediante las diferentes herramientas heurísticas se mejora a los siguientes objetivos:

- ✓ *Objetivo 4. "Educación de calidad"*. Mediante la mejora de la adecuación del número de alumnos a las aulas se fomenta a recibir una educación de mejor calidad, realizando los exámenes en ambientes favorables para su correcto funcionamiento.
- ✓ *Objetivo 8. "Trabajo decente y crecimiento económico"*. Al reducir el número de clases empleadas, en consecuencia, se emplean un menor número de docentes para su vigilancia por lo que se fomenta a que sea un trabajo creciente y a que más docentes puedan emplear tiempo en labores divulgativas.
- ✓ *Objetivo 9. "Industria, innovación e infraestructura"*. Con la realización de un trabajo de investigación de este calibre se fomenta a la innovación proponiendo un modelo novedoso para la programación de exámenes.
- ✓ *Objetivo 12. "Producción y consumo responsables"*. Como se comentaba anteriormente, al disminuir la utilización de aulas, se reduce en consecuencia una reducción de la utilización de recursos energéticos.

En conclusión, mediante la implementación de las herramientas se tiene como objetivo fomentar la igualdad entre todos los estudiantes, reducir los gastos en energía, dinero, mantenimiento y personal, y fomentar el uso eficiente de los recursos y fomentar la conciencia y práctica sostenible.

7. FUTURAS LÍNEAS DE ACTUACIÓN Y CONCLUSIONES

7.1. Futuras líneas de actuación

Como se ha comentado en anteriores capítulos, la investigación en la línea del ETP es amplia en la actualidad. En el caso en particular abordado en este TFG, se plantean dos líneas de actuación que completarían las propuestas de este trabajo, partiendo de la solución implementada de la heurística constructiva.

En un primer lugar, se podría realizar un algoritmo genético basado en la heurística constructiva del apartado 4, en ella cada solución o cromosoma representaría una asignación de exámenes a aulas y horarios y los genes de este cromosoma podrían ser pares examen-aula. La función de evaluación que se elegiría consideraría el cumplimiento de las restricciones duras planteadas en este TFG y minimizaría la distancia entre las aulas asignadas a los exámenes. Los operadores genéticos utilizados que se pueden tomar:

- Selección. Utilizar un método de selección como la ruleta o el torneo para elegir soluciones padres basadas en su aptitud.
- Cruce. Se aplica operadores de cruce que respeten la estructura de las soluciones, asegurando que las nuevas soluciones resultantes sean válidas y factibles.
- Mutación. Se implementa operadores de mutación que introduzcan pequeños cambios aleatorios en la asignación de exámenes a aulas y horarios, asegurándose de mantener la factibilidad de la solución.

Por último, se definiría parámetros de tamaño de la población, la tasa de cruce, la tasa de mutación y el número de generaciones, ajustando los parámetros según la complejidad que se proponga en el problema y los recursos disponibles. Entre los beneficios que se sacarían de la aplicación de un algoritmo de este tipo se destaca:

- Mejora continua. A través de las operaciones de cruce y mutación, el algoritmo genético puede explorar nuevas soluciones más allá de las generadas por la heurística constructiva inicial.
- Diversidad de soluciones. La diversidad genética mantenida por la selección y los operadores genéticos permite explorar diferentes configuraciones de asignación de exámenes y aulas.
- Optimización adaptativa. A medida que avanza el algoritmo genético, la población tiende a converger hacia soluciones óptimas o cercanas al óptimo global del problema.

Por otro lado, se toma como segunda propuesta, un Algoritmo de búsqueda tabú. Se podría realizar un algoritmo de búsqueda tabú basado en la heurística constructiva, en ella se define la forma en la que se generarían los estados vecinos, representando un estado una asignación de exámenes a aulas y horarios. Los movimientos para generar vecinos podrían incluir intercambios de aulas entre exámenes o cambios de días, asegurándose de mantener la factibilidad de las soluciones.

La función de evaluación que se elegiría puede calcular la aptitud de cada solución, considerando la minimización de la distancia entre las aulas asignadas e implementa un mecanismo tabú para evitar ciclos y promover la exploración del espacio de soluciones. Esto puede implicar mantener una lista tabú que registre los movimientos recientes y que prohíba revertir cambios recientes o realizar movimientos similares.

Se puede considerar como método de parada un número máximo de iteraciones como el aplicado en este TFG. Los beneficios que podría proporcionar esta metaheurística serían:

- Exploración eficiente. La búsqueda tabú permite explorar rápidamente diferentes vecindarios de soluciones, buscando mejoras sin caer en ciclos o soluciones subóptimas.
- Adaptabilidad. Puede ajustar la estrategia tabú para enfocarse en diferentes aspectos del problema, como la minimización de distancias entre aulas.
- Capacidad de encontrar soluciones robustas. Al integrar la heurística constructiva, se comienza con una solución de alta calidad y refinada que actúa como punto de partida sólido para la búsqueda tabú.

7.2. Conclusiones finales

Durante la progresión de este TFG se realiza la propuesta de dos herramientas de apoyo a la programación del calendario de exámenes. Una primera herramienta de tipo heurística, basada en el tipo constructivo y una heurística del tipo de los Algoritmos de búsqueda local. El primer modelo heurístico pretende la asignación de las asignaturas en el calendario de exámenes basándose en el mayor número de alumnos, tanto en los matriculados en el caso de las asignaturas de los diferentes grados como en la capacidad de las aulas disponibles para albergar las pruebas; por otro lado, se tiene en cuenta el factor de la holgura entre la capacidad del aula asignada y el número de alumnos matriculados para las asignaturas designadas en un único aula; y el factor de las distancias entre las aulas asignadas a las asignaturas cuyo número de matriculados sobre pasa la capacidad de las aulas, permitiendo así una mejora considerable en la asignación. En segundo lugar, como complemento a la solución óptima obtenida con la anterior heurística se aborda un algoritmo de búsqueda local que, partiendo de la solución inicial realizan la permutación de una asignatura de forma aleatoria y que mejora la distancia entre las aulas asignadas para ese nuevo día seleccionado.

Para la realización de la propuesta, se aborda una explicación de las diferentes heurísticas propuestas y, sobre todo de la problemática conocida como ETP (Educational Timetabling Problem). Como complemento se describe el entorno en el que se pone en práctica las herramientas, la Escuela Técnica Superior de Ingeniería Industrial (ETSII) de la UPV.

En un tercer lugar, se justifica la realización de los modelos heurísticos mediante las líneas antecedentes de actuación que abordan la problemática descrita. Al comparar los elementos y dimensiones que se evaluaron en cada uno de los estudios, se puede ver que ninguno de ellos incluye todos los factores que se consideraron en las propuestas desarrolladas a través de este TFG, sino que se realiza una mezcla de los diferentes propuestos, lo cual detalla la originalidad de la propuesta.

A continuación, se describe la problemática, tanto en el Capítulo 4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS como con el Capítulo 5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS. En el primero de estos, se presenta una herramienta heurística para mejorar la programación de exámenes en una Escuela o Facultad. Se analizan las deficiencias del método actual de la ETSII, que es manual, inflexible y no optimiza recursos, resultando en solapamientos y uso ineficiente de personal y aulas.

La nueva propuesta tiene como objetivo reducir la distancia entre las aulas asignadas para los exámenes, optimizando así el uso de recursos y facilitando la logística. La heurística constructiva utiliza un enfoque devorador, priorizando la asignación de las asignaturas con mayor número de matriculados a las aulas de mayor capacidad y cercanas entre sí.

El algoritmo se implementa en varios pasos: ordena asignaturas y aulas, asigna las más grandes primero y ajusta según capacidad y distancia. Esto se realiza en Microsoft Excel y VBA, permitiendo una fácil manipulación de datos y una interfaz amigable. La herramienta genera automáticamente un calendario de exámenes optimizado. La heurística constructiva desarrollada proporciona una solución innovadora y eficaz para la planificación de exámenes en la ETSII, mejora la eficiencia operativa y la experiencia académica, y es adaptable a futuras modificaciones. Esta metodología puede aplicarse a otras instituciones, mostrando su potencial como una solución integral para la gestión de calendarios de exámenes.

En el Capítulo 5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS, se aborda el desarrollo y la implementación de un Algoritmo de Búsqueda Local como herramienta heurística para optimizar la programación de exámenes en una Escuela o Facultad. El problema principal consiste en coordinar múltiples exámenes de distintos grados y cursos a lo largo de dos cuatrimestres, respetando reglas específicas como turnos de mañana y tarde según el curso y evitando solapamientos entre exámenes de diferentes años. El modelo de búsqueda local se centra en minimizar la distancia total entre las aulas asignadas para cada examen. Se emplea la heurística constructiva del anterior capítulo que inicia con una solución inicial y luego explora soluciones vecinas mediante cambios aleatorios en la asignación de aulas y turnos, asegurando cumplir con restricciones duras y blandas. Se evalúa la función objetivo basada en la distancia total entre aulas, adoptando la mejor solución encontrada. El modelo utiliza una estructura de datos en MS Excel y están programados en Visual Basic, lo que facilita la manipulación de datos y la generación del horario optimizado. En resumen, este capítulo ofrece un enfoque práctico y eficiente para mejorar la programación de exámenes mediante el uso de heurísticas de búsqueda local. La implementación detallada proporciona una solución que optimiza el uso de recursos y minimiza la distancia entre aulas, cumpliendo con las complejas restricciones académicas y operativas, lo que contribuye significativamente a la eficiencia y calidad de la planificación académica.

Para terminar, se realiza una comparación de las diferentes heurísticas a modo de caracterización de que solución de las propuestas es mejor, o cuál de ellas propone una solución más factible a modo de análisis de las diferentes soluciones propuestas. Además, en ese apartado se tratan nuevas líneas de actuación que podrían mejorar aún más las soluciones óptimas.

BIBLIOGRAFÍA

- Abdipoor, S., Yaakob, R., Goh, S., & Abdullah, S. (2023). Meta-heuristic approaches for the University Course Timetabling Problem. *Elsevier*, 4.
- Álvarez-Valdés, R., Crespo, E., & Tamarit, J. (1997). A tabu search algorithm to schedule university examinations. *Qüestió*, 201-215.
- Andrés Romano, C. (2018). *OPTIMIZACIÓN METAHEURÍSTICA PARA INGENIEROS. APLICACIONES EN ORGANIZACIÓN INDUSTRIAL*. Valencia: DELTA Publicaciones.
- Astaiza A., L. G. (Diciembre de 2005). Programación de exámenes: un enfoque práctico. *Revista Ingeniería e Investigación*, 25(3), 94.
- Ceschia, S., Di Gaspero, L., & Schaerf, A. (2022). Educational timetabling: Problems, benchmarks, and state-of-the-art . 2.
- Ekanayake, T., Subasinghe, P., Ragel, S., Gamage, A., & Attanayaka, S. (2019). Intelligent Timetable Scheduler: A Comparison of Genetic, Graph Coloring, Heuristic and Iterated Local Search Algorithms . Malabe, Sri Lanka.
- ETSII-UPV. (2024). Obtenido de <https://www.etsii.upv.es/docencia/titulaciones-es.php>
- ETSII-UPV. (2024). Obtenido de <https://www.etsii.upv.es/presentacion/presentacion-es.php#:~:text=Alrededor%20de%204.100%20alumnos%20matriculados,los%20Centros%20de%20la%20UPV.>
- Gashgari, R., Alhashimi, L., Aljawi, L., Alamoudi, A., Obaid, D., & Palaniswamy, D. (2018). A SURVEY ON EXAM SCHEDULING. *IEEE*.
- Kahar, M., & Kendall, G. (2014). Universiti Malaysia Pahang examinationtimetabling problem: scheduling invigilators. *Journal of the Operational Research Society*.
- Melian, B., Moreno Pérez, J., & Moreno Vega, J. (2003). Metaheurísticas: una visión global. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*(19), 7-28.
- ONU. (2024). <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>.
- Siew, E., Sze, S., Goh, S., Kendall, G., Sabar, N., & Abdullah, S. (2022 de Marzo de 2024). A Survey of Solution Methodologies for. *IEEE*.
- Soria-Alcaraz, J., Özcan, E., Swan, J., Kendall, G., & Carpio, M. (2016). Iterated local search using an add and delete hyper-heuristic for. *Applied Soft Computing*.
- Universitat Politècnica de València. (25 de Abril de 2024). Obtenido de <https://www.upv.es/entidades/vecal/proyecto-de-actualizacion-de-competencias-transversales/>
- UPV. (2024). Obtenido de <https://www.upv.es/organizacion/la-institucion/historia/index-es.html>
- Vicens Salort, E., Ortiz Bas, Á., & Guarch Bertolín, J. (1997). *MÉTODOS CUANTITATIVOS VOLUMEN I*.

Wibowo, A., Teoh, C., & Ngadiman, M. (1 de Febrero de 2013). Review of state of the art for metaheuristic techniques. *Springer Science+Business Media Dordrecht*, 2.

ANEXO

Anexo I. Código de la heurística constructiva

```

Dim Au As Integer          'número de aulas
Dim Asi As Integer        'número de asignaturas
Dim AsEq As Integer      'número de parejas en listado asignaturas
equivalentes
Dim AuPar As Integer      'número de parejas de aulas
Dim Aux1 As Integer      'número de filas en Compatibilidad (conjunto de
titulación, curso, día y vía)

Dim Datos_Asignaturas() As Long      'matriz con información de asignaturas
(titulación, curso, cuatrimestre, asignatura, matriculados, vía)
Dim Cod_Asig_Eq() As Long            'matriz con combinación asignaturas
equivalentes (asignatura1, asignatura2)
Dim Datos_Capacidad() As Long      'matriz con información de aulas (aula,
capacidad normal, capacidad examen)
Dim Datos_Distancias() As Long     'matriz con distancias entre aulas (aula 1,
aula 2, distancia)
Dim Disponibilidad() As Integer    'matriz sobre disponibilidad de las aulas
(fila de aula, día). Tiene tantas filas como aulas * número de turnos. En la
primera columna está guardada el aula, en las siguientes, los días.
Dim Horario_Solucion() As String   'matriz para almacenar la solución (fila de
aula, día). como la de Disponibilidad pero no tiene en la primera columna las
aulas.
Dim Compatibilidad() As Long       'matriz sobre compatibilidad (titulación,
curso, día, compatibilidad, vía). Valor 0 si ese curso, titulación y vía aun no
tiene asignado examen ese día, o 1 si si que tiene asignado un examen ese día
Dim asignada() As Long            'matriz sobre si una asignatura está asignada
o no a una fecha de examen.
Dim Optativas() As Long          'matriz con listado de asignaturas optativas.
Dim capacidadtotal() As Integer   'matriz que almacena la capacidad total
disponible en cada turno y día (turno, día)

Dim numdias As Integer          'número de días en los que se puede programar
examen
Dim cuatrimestre As Integer     'indica el turno que se quiere programar (1: A,
2: B)
Dim turno As Integer           'indica el turno que se quiere programar (1: ma-
ana, 2: tarde, 0: ambos)
Dim día As Integer
Dim Aula As Integer
Dim aux As Integer
Dim NOpt As Integer
Dim PosOpt As Integer
Dim TurnoAs As Integer

```

```
Dim Asignatura As Long
Dim sheetWork As String
Dim Col As Integer
Dim fila As Integer
```

```
Dim nombre As String
Dim AsigEq() As Long
```

```
.....
'Lectura de datos directos
.....
'Hay tres datos que se le tienen que proporcionar a la heurística
'De manera que el usuario lo escriba en la Excel antes de ejecutar la heurística
'Estos datos son: el número de días, el cuatrimestre, y el turno.
numdias = Worksheets("HEURÍSTICA").Cells(7, 3).Value
cuatrimestre = Worksheets("HEURÍSTICA").Cells(5, 3).Value
turno = Worksheets("HEURÍSTICA").Cells(6, 3).Value
```

```
.....
'Leer tablas de datos
.....
Call Calcular_Tamaño("DATOS CAPACIDAD", Au, Col)
Call Leer_Tabla_Col("DATOS CAPACIDAD", Au, Col, Datos_Capacidad())

Call Calcular_Tamaño("DATOS DISTANCIAS", AuPar, Col)
Call Leer_Tabla_Col("DATOS DISTANCIAS", AuPar, Col, Datos_Distancias())

Call Calcular_Tamaño("DATOS ASIGNATURAS COD", Asi, Col)
Call Leer_Tabla_Col("DATOS ASIGNATURAS COD", Asi, Col, Datos_Asignaturas())

Call Calcular_Tamaño("CÓD. ASIG. EQ.", AsEq, Col)
Call Leer_Tabla_Col("CÓD. ASIG. EQ.", AsEq, Col, Cod_Asig_Eq())
ReDim AsigEq(AsEq)

Call Calcular_Tamaño("COMPATIBILIDAD", Aux1, Col)
Call Leer_Tabla_Col("COMPATIBILIDAD", Aux1, Col, Compatibilidad())
```



```
Call Calcular_Tamaño("OPTATIVAS", NOpt, Col)
```

```
Call Leer_Tabla_Col("OPTATIVAS", NOpt, Col, Optativas())
```

'Por otra parte habría que definir una función diferente para la lectura de tablas que tienen rotulos tanto en la primera fila como en la primera columna

'Entiendo que es la función que tienes puesta como Leer_Disponibilidad.

```
Call Leer_Disponibilidad("DATOS DISPONIBILIDAD", Au, turno, numdias + 1, Disponibilidad())
```

```
.....
```

```
''' CREACION DE LA MATRIZ SOLUCION '''
```

```
.....
```

```
If turno = 1 Or turno = 2 Then
```

```
    ReDim Horario_Solucion(Au, numdias)
```

```
ElseIf turno = 0 Then
```

```
    ReDim Horario_Solucion(2 * Au, numdias)
```

```
End If
```

```
.....
```

```
''' ORDENAR CLASES Y ASIGNATURAS DE MAYOR A MENOR CAPACIDAD/MATRICULADOS '''
```

```
.....
```

```
Call Ordenar_Matriz(Datos_Capacidad(), 3)
```

```
Call Ordenar_Matriz(Datos_Asignaturas(), 5)
```

```
.....
```

```
''' IDENTIFICAR ASIGNATURAS A PROGRAMAR '''
```

```
.....
```

'Actualizamos el vector de asignada() para que solamente se tengan en cuenta las asignaturas del turno y cuatrimestre que tocan:

'En la primera columna pongo el número de la asignatura y en la segunda el valor binario

```
ReDim asignada(Asi, 2)
```

```
For x = 1 To Asi
```

```
    asignada(x, 1) = Datos_Asignaturas(x, 4)
```

```
    If (Datos_Asignaturas(x, 3) = cuatrimestre Or Datos_Asignaturas(x, 3) = 0) Then 'Comprobamos que la asignatura se imparte en el cuatrimestre que vamos a programar o es anual.
```

```

    If turno = 1 Then 'Si es turno de mañana comprobamos que sea asignatura
de 1° o 3°, en caso contrario le ponemos que ya está asignada para que no la
tenga en cuenta.

```

```

        If (Datos_Asignaturas(x, 2) = 1 Or Datos_Asignaturas(x, 2) = 3) Then
            asignada(x, 2) = 0
        ElseIf (Datos_Asignaturas(x, 2) = 2 Or Datos_Asignaturas(x, 2) = 4)
Then
            asignada(x, 2) = 1
        End If

```

```

    ElseIf turno = 2 Then 'Lo contrario para cuando sea el turno de tarde
        If (Datos_Asignaturas(x, 2) = 1 Or Datos_Asignaturas(x, 2) = 3) Then
            asignada(x, 2) = 1
        ElseIf (Datos_Asignaturas(x, 2) = 2 Or Datos_Asignaturas(x, 2) = 4)
Then
            asignada(x, 2) = 0
        End If

```

```

    ElseIf turno = 0 Then 'Si se desea programar todos los turnos (mañana y
tarde) entonces todas las asignaturas del cuatri se deben poner como pendientes
de asignar.

```

```

        asignada(x, 2) = 0
    End If
    Else 'En caso contrario, la asignatura se imparte en un cuatri que no
queremos programar y por tanto, no se debe asignar
        asignada(x, 2) = 1
    End If

```

```

Next x

```

```

.....
''' IDENTIFICACIÓN CAPACIDAD TOTAL DIARIA '''
.....

```

```

'Necesito para hacer más fácil la asignación un matriz (turno, dias) donde se
guarde por cada día la capacidadtotal disponible teniendo en cuenta las aulas
que no se han utilizado aún

```

```

ReDim capacidadtotal(2, numdias)

```

```

For x = 1 To Au

```

```

    For y = 1 To numdias

```

```

        capacidadtotal(1, y) = capacidadtotal(1, y) + Datos_Capacidad(x, 3)

```

```

        capacidadtotal(2, y) = capacidadtotal(2, y) + Datos_Capacidad(x, 3)

```

```

    Next y

```

```

Next x

```

```

.....
''' ASIGNACIÓN AULA Y DÍA '''
.....

'Lo primero que hay que hacer es comprobar de asignatura con mayor a menor
matriculados si cabe en el aula con mayor a menor capacidad

'Hay que coger la asignatura con más alumnos y ver si hay que programar su examen
o no:

For x = 1 To Asi
  Asignatura = Datos_Asignaturas(x, 4)
  If asignada(x, 2) = 0 Then 'Si la asignatura está por asignar

      .....
      ''' IDENTIFICAR TURNO ASIGNATURA '''
      .....

      'Identificar el turno en que se debe hacer la asignatura. Si es de 1 o
      3 curso, por la ma-ana. Si es de 2 o 4 curso, por la tarde.

      'A no ser que sea optativa, que en ese caso, se haría en el turno opuesto
      PosOpt = 0

      If Datos_Asignaturas(x, 2) = 1 Or Datos_Asignaturas(x, 2) = 3 Or
      Datos_Asignaturas(x, 2) = 5 Then
        TurnoAs = 1
        For y = 1 To NOpt
          If Asignatura = Optativas(y, 1) Then
            TurnoAs = 2
            PosOpt = y
            y = NOpt
          End If
        Next y
      ElseIf Datos_Asignaturas(x, 2) = 2 Or Datos_Asignaturas(x, 2) = 4 Then
        TurnoAs = 2
        For y = 1 To NOpt
          If Asignatura = Optativas(y, 1) Then
            TurnoAs = 1
            PosOpt = y
            y = NOpt
          End If
        Next y
      End If
End If

```

```

.....
''' COMPROBAR SI ASIGNATURA TIENE EQUIVALENTES '''
.....

ContadorAsEq = 0
MatriculadosEq = Datos_Asignaturas(x, 5)
nombre = Asignatura
For i = 1 To AsEq
    If Cod_Asig_Eq(i, 1) = Asignatura Then
        AsignaturaEq = Cod_Asig_Eq(i, 2)
        ContadorAsEq = ContadorAsEq + 1
        AsigEq(ContadorAsEq) = Cod_Asig_Eq(i, 2)
        nombre = nombre & ";" & AsignaturaEq 'El nombre que tenga hasta
el momento le añade ";" y el nombre de la nueva asignatura equivalente
        For j = 1 To Asi
            If AsignaturaEq = Datos_Asignaturas(j, 4) Then
                MatriculadosEq = MatriculadosEq + Datos_Asignaturas(j,
5)
                    j = Asi
            End If
        Next j
    End If
Next i
For j = ContadorAsEq + 1 To AsEq
    AsigEq(j) = 0
Next j

'Al acabar este bucle, en matal está el número de matriculados, en
ContadorAsEq el número de asignaturas equivalentes,

'y en AsigEq() el listado de asignaturas equivalentes a la asignatura
Asignatura

.....
.....
''' COMPROBAR EL PRIMER DIA EN QUE SE PUEDE PROGRAMAR ASIGNATURA Y
EQUIVALENTES '''
.....
.....

dia = 0
capacidadmax = 0

```

```

For t = 1 To numdias
    compatible = 0

    If PosOpt = 0 Then ' Si la asignatura es normal (es decir, no es una
    optativa). Esto se comprueba porque en el caso de optativas, no tienen
    incompatibilidades (se pueden hacer varias optativas el mismo día ya que el
    alumno usualmente escoge una).

        'Primero miro si la "Asignatura" se puede hacer en ese día

        For i = 1 To Aux1

            If Compatibilidad(i, 1) = Datos_Asignaturas(x, 1) And
            Compatibilidad(i, 2) = Datos_Asignaturas(x, 2) And Compatibilidad(i, 3) = t And
            Compatibilidad(i, 5) = Datos_Asignaturas(x, 6) Then

                compatible = compatible + Compatibilidad(i, 4)

                i = Aux1

            End If

        Next i

        'Luego compruebo si sus equivalentes se pueden hacer en ese día
        If ContadorAsEq > 0 Then

            For y = 1 To ContadorAsEq

                For k = 1 To Asi 'Localizo en que fila de
                Datos_Asignaturas está la asignatura equivalente (PosEq)

                    If Datos_Asignaturas(k, 4) = AsigEq(y) Then

                        PosEq = k

                        k = Asi

                    End If

                Next k

                For i = 1 To Aux1

                    If Compatibilidad(i, 1) = Datos_Asignaturas(PosEq,
                    1) And Compatibilidad(i, 2) = Datos_Asignaturas(PosEq, 2) And Compatibilidad(i,
                    3) = t And Compatibilidad(i, 5) = Datos_Asignaturas(PosEq, 6) Then

                        compatible = compatible + Compatibilidad(i, 4)

                        i = Aux1

                    End If

                Next i

            Next y

        End If

    End If

    'En caso de que en este punto el contador de compatible sea igual a
    0 querría decir que todas las asignaturas equivalentes también son compatibles,
    'por lo que nos guardamos ese día para hacer la programación.

    'Siempre y cuando, quepan los matriculados en la asignatura y
    equivalentes

```

'Le voy a añadir que compruebe si es el día con mayor capacidad disponible, así intentamos que no se llenen los primeros días y se arregle el problema de que hay

'asignaturas sin ningún día compatible.

```
If compatible = 0 And capacidadtotal(TurnoAs, t) >= MatriculadosEq
And capacidadtotal(TurnoAs, t) > capacidadmax Then
```

```
    If capacidadtotal(TurnoAs, t) > 0 Then 'Esto ha sido
    necesario ponerlo porque sino quería asignar asignatura sin matriculados a día
    que no tiene aulas libres. y igualmente hay que asignar aula...
```

```
        dia = t
```

```
        't = numdias
```

```
        capacidadmax = capacidadtotal(TurnoAs, t)
```

```
    End If
```

```
End If
```

```
Next t
```

'Al acabar esto tenemos guardado en dia el día en el que queremos programar

'Si aquí día fuera 0, tendríamos un problema. Querría decir que no ha encontrado ningún día en que poder hacer la asignatura y sus equivalentes.

'Esperemos que esto no ocurra.

```
.....
```

```
''' ASIGNACIÓN DE AULA '''
```

```
.....
```

```
For y = 1 To Au
```

```
    Aula = Datos_Capacidad(y, 1)
```

'Identificar la posición del aula en la matriz disponibilidad

```
    For j = 1 To Au
```

```
        If Aula = Disponibilidad(j, 1) Then
```

```
            Posaula = j
```

```
            j = Au
```

```
        End If
```

```
        If TurnoAs = 2 Then
```

```
            Posaula = Posaula + Au
```

```
        End If
```

```
    Next j
```

'Comprobar que el aula está disponible en el día

```
If Disponibilidad(Posaula, dia + 1) = 0 Then
```

MatriculadosEqaux = MatriculadosEq 'Creo esta variable para poder ir actualizando el valor y no perder lo de MatriculadosEq

'Comprobar si caben los alumnos en una sola aula, y asignarsela

```

If MatriculadosEq <= Datos_Capacidad(y, 3) Then
    'Antes de asignarle el aula, miro si hay otra más pequeña
    que le pueda venir bien, para dejar libres las grandes
    mejorcap = Datos_Capacidad(y, 3)
    PosaulaC = y
    If TurnoAs = 1 Then
        idt = 0
    ElseIf TurnoAs = 2 Then
        idt = Au
    End If
    For i = y + 1 To Au
        For j = 1 + idt To Au + idt
            If Datos_Capacidad(i, 1) = Disponibilidad(j, 1) And
Disponibilidad(j, dia + 1) = 0 Then
                If MatriculadosEq <= Datos_Capacidad(i, 3) And
mejorcap > Datos_Capacidad(i, 3) Then
                    PosaulaC = i
                    Posaula = j
                    mejorcap = Datos_Capacidad(i, 3)
                End If
            End If
        Next j
    Next i
    Horario_Solucion(Posaula, dia) = nombre
    Disponibilidad(Posaula, dia + 1) = 1
    capacidadtotal(TurnoAs, dia) = capacidadtotal(TurnoAs, dia)
- Datos_Capacidad(PosaulaC, 3)
    MatriculadosEqaux      =      MatriculadosEqaux      -
Datos_Capacidad(PosaulaC, 3)
    asignada(x, 2) = 1 'La asignatura x ya se ha programado
    If ContadorAsEq > 0 Then 'Bucle para asignar las
equivalentes (decir que ya se ha programado su examen)
        For i = 1 To ContadorAsEq
            For j = 1 To Asi
                If Datos_Asignaturas(j, 4) = AsigEq(i) Then
                    asignada(j, 2) = 1
                    j = Asi
                End If
            Next j
        Next i
    End If

```

y = Au 'Para que salga del "For y" y se vaya a la siguiente asignatura

'En el caso en que no quepan, hay que asignar el aula inicial que hemos seleccionado ya, y también las aulas más cercanas hasta cubrir los alumnos

```

ElseIf MatriculadosEq > Datos_Capacidad(y, 3) Then
    Horario_Solucion(Posaula, dia) = nombre
    Disponibilidad(Posaula, dia + 1) = 1
    capacidadtotal(TurnoAs, dia) = capacidadtotal(TurnoAs, dia)
- Datos_Capacidad(y, 3)
    MatriculadosEqaux = MatriculadosEq - Datos_Capacidad(y, 3)
    asignada(x, 2) = 1
    If ContadorAsEq > 0 Then
        For i = 1 To ContadorAsEq
            For j = 1 To Asi
                If Datos_Asignaturas(j, 4) = AsigEq(i) Then
                    asignada(j, 2) = 1
                    j = Asi
                End If
            Next j
        Next i
    End If
    'Hasta aquí es igual que el caso en que cabe en un aula.
    ahora vamos a ver que hacer con el resto de alumnos:
    'Ahora vamos a proceder a asignar otras aulas, hasta que se
    complete el número de matriculados
    Do While MatriculadosEqaux > 0
        min = 10000
        For i = 1 To AuPar
            If Datos_Distancias(i, 1) = Aula Then
                Aula2 = Datos_Distancias(i, 2)
                'Localizamos en que fila de la tabla
disponibilidad está el aula2
                For k = 1 To Au
                    If Aula2 = Disponibilidad(k, 1) Then
                        If TurnoAs = 1 Then
                            PosaulaD = k
                            k = Au
                        ElseIf TurnoAs = 2 Then
                            PosaulaD = k + Au
                            k = Au
                        End If
                    End If
                End If
            End If
        Next i
    End Do

```



```

        End If
    Next k
    'Localizamos en que fila de la tabla capacidades
está el aula2
    For k = 1 To Au
        If Aula2 = Datos_Capacidad(k, 1) Then
            PosaulaC = k
            k = Au
        End If
    Next k
    'En el caso que tenga menor distancia, y este
disponible, se guarda el nuevo valor mínimo de distancia y el aula (en mejor)
    If Disponibilidad(PosaulaD, dia + 1) = 0 And
Datos_Distancias(i, 3) < min Then
        min = Datos_Distancias(i, 3)
        PosaulaDmejor = PosaulaD
        PosaulaCmejor = PosaulaC
        aulamejor = Aula2
    ElseIf Disponibilidad(PosaulaD, dia + 1) = 0 And
Datos_Distancias(i, 3) = min Then
        'En el caso de empate por distancia entre
Aula2 (nueva aula candidata) y aulamejor (mejor aula candidata hasta el momento)
        'Se escoge la que mejor se ajuste a los
alumnos que quedan por ubicar
        If Abs(Datos_Capacidad(PosaulaC, 3) -
MatriculadosEqaux) < Abs(Datos_Capacidad(PosaulaCmejor, 3) - MatriculadosEqaux)
Then
            min = Datos_Distancias(x, 3)
            PosaulaDmejor = PosaulaD
            PosaulaCmejor = PosaulaC
            aulamejor = Aula2
        End If
    End If
End If
End If
Next i
'Como resultado de este for tendré un aula seleccionada
para completar la solución.
'Ahora actualizo la solución con esta aula
Horario_Solucion(PosaulaDmejor, dia) = nombre
Disponibilidad(PosaulaDmejor, dia + 1) = 1
MatriculadosEqaux = MatriculadosEqaux -
Datos_Capacidad(PosaulaCmejor, 3)

```

```

        capacidadtotal(TurnoAs, dia) = capacidadtotal(TurnoAs,
dia) - Datos_Capacidad(PosaulaCmejor, 3)
        Loop
        y = Au
    End If
End If
Next y

.....
''' ACTUALIZAR COMPATIBILIDAD DE LAS TITULACIONES Y CURSOS '''
.....

'Solo hay que hacer esto de actualizar la compatibilidad, si las
asignaturas no son optativas.

'Esto se debe a que si son optativas se pueden solapar sin problema.
If PosOpt = 0 Then
    If ContadorAsEq = 0 Then
        For i = 1 To Aux1
            If Compatibilidad(i, 1) = Datos_Asignaturas(x, 1) And
Compatibilidad(i, 2) = Datos_Asignaturas(x, 2) And Compatibilidad(i, 3) = dia
And Compatibilidad(i, 5) = Datos_Asignaturas(x, 6) Then
                Compatibilidad(i, 4) = 1
                i = Aux1
            End If
        Next i
    End If
    If ContadorAsEq > 0 Then
        For i = 1 To Aux1
            If Compatibilidad(i, 1) = Datos_Asignaturas(x, 1) And
Compatibilidad(i, 2) = Datos_Asignaturas(x, 2) And Compatibilidad(i, 3) = dia
And Compatibilidad(i, 5) = Datos_Asignaturas(x, 6) Then
                Compatibilidad(i, 4) = 1
                i = Aux1
            End If
        Next i
        For y = 1 To ContadorAsEq
            For k = 1 To Asi 'Localizo en que fila de Datos_Asignaturas
está la asignatura equivalente (PosEq)
                If Datos_Asignaturas(k, 4) = AsigEq(y) Then
                    PosEq = k
                    k = Asi
                End If
            End If
        Next y
    End If
End If

```

```

Next k
For i = 1 To Aux1
    If Compatibilidad(i, 1) = Datos_Asignaturas(PosEq, 1)
And Compatibilidad(i, 2) = Datos_Asignaturas(PosEq, 2) And Compatibilidad(i, 3)
= dia And Compatibilidad(i, 5) = Datos_Asignaturas(PosEq, 6) Then
        Compatibilidad(i, 4) = 1
        i = Aux1
    End If
Next i
Next y
End If
End If
End If
Next x

```

```

.....
''' IMPRIMIR SOLUCIÓN '''
.....
For x = 1 To 2 * Au
    For y = 1 To numdias
        Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(x + 2, y + 17).Value =
Horario_Solucion(x, y)
    Next y
Next x

```

```

.....
''' EVALUACIÓN SOLUCIÓN '''
.....
distanciatotal = 0
If turno = 0 Then
    nfil = 2 * Au
Else
    nfil = Au
End If
For t = 1 To numdias
    For i = 1 To nfil - 1
        For j = i + 1 To nfil
            If Horario_Solucion(i, t) <> "" Then

```

If Horario_Solucion(i, t) = Horario_Solucion(j, t) Then 'Si coincide la asignatura programada en dos aulas hay que tener en cuenta su distancia.

```

        Aula1 = Disponibilidad(i, 1)
        Aula2 = Disponibilidad(j, 1)
        For x = 1 To AuPar
            If Aula1 = Datos_Distancias(x, 1) And Aula2 =
Datos_Distancias(x, 2) Then
                distanciatotal = distanciatotal +
Datos_Distancias(x, 3)
            End If
        Next x
    End If
End If
Next j
Next i
Next t
Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(3, 29).Value = distanciatotal

```

holguratotalsolucion = 0

capacidadtotalsolucion = 0

matriculadostotalsolucion = 0

If turno = 0 Then

numturnos = 2

Else

numturnos = 1

End If

For x = 1 To Au

For y = 1 To Au * numturnos

If Datos_Capacidad(x, 1) = Disponibilidad(y, 1) Then

For t = 1 To numdias

capacidadtotalsolucion = capacidadtotalsolucion +
Datos_Capacidad(x, 3) * Disponibilidad(y, t + 1)

Next t

End If

Next y

Next x

For x = 1 To Asi

If cuatrimestre = Datos_Asignaturas(x, 3) Then

matriculadostotalsolucion = matriculadostotalsolucion +
Datos_Asignaturas(x, 5)

```
End If
Next x
holguratotalsolucion = capacidadtotalsolucion - matriculadostotalsolucion
Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(6, 29).Value = holguratotalsolucion

End Sub
```

Anexo II. Código de la función Calcular_Tamaño

```
Sub Calcular_Tamaño(SheetName As String, ByRef numFil As Integer, ByRef numCol
As Integer)
    'En esta función se asume que en cada hoja solo está disponible la
información que queremos leer, y que esta empieza en la fila 2, columna 1 (ya
que la primera fila es de rotulos).
    numFil = 0
    numCol = 0

    Do While Worksheets(SheetName).Cells(2 + numFil, 1).Value <> ""
        numFil = numFil + 1
    Loop

    Do While Worksheets(SheetName).Cells(1, 1 + numCol).Value <> ""
        numCol = numCol + 1
    Loop

End Sub
```

Anexo III. Código de la función Leer_Tabla_Col y Leer_Disponibilidad

```
Sub Leer_Tabla_Col(SheetName As String, ByVal numFil As Integer, ByVal numCol
As Integer, ByRef matrizRes() As Long)
```

```
    ReDim matrizRes(numFil, numCol)
```

```
    For i = 1 To numFil
```

```
        For j = 1 To numCol
```

```
            matrizRes(i, j) = Worksheets(SheetName).Cells(i + 1, j).Value 'La
fila es i+1 porque hemos dicho que en la primera fila siempre tendremos rotulo.
```

```
        Next j
```

```
    Next i
```

```
End Sub
```

```
Sub Leer_Disponibilidad(SheetName As String, ByVal naulas As Integer, ByVal
nturnos As Integer, ByVal ndias As Integer, ByRef matrizRes2() As Integer)
```

```
    If nturnos = 1 Or nturnos = 2 Then
```

```
        ReDim matrizRes2(naulas, ndias + 1)
```

```
    ElseIf nturnos = 0 Then
```

```
        ReDim matrizRes2(2 * naulas, ndias + 1)
```

```
    End If
```

```
    If nturnos = 0 Then
```

```
        For x = 1 To 2 * naulas
```

```
            For y = 1 To ndias + 1
```

```
                matrizRes2(x, y) = Worksheets(SheetName).Cells(x + 1, y +
1).Value
```

```
            Next y
```

```
        Next x
```

```
    ElseIf nturnos = 1 Then
```

```
        For x = 1 To naulas
```

```
            For y = 1 To ndias + 1
```

```
                matrizRes2(x, y) = Worksheets(SheetName).Cells(x + 1, y +
1).Value
```

```
            Next y
```

```
        Next x
```

```
    ElseIf nturnos = 2 Then
```

```
        For x = 1 To naulas
```

```
            For y = 1 To ndias + 1
```

```
        matrizRes2(x, y) = Worksheets(SheetName).Cells(x + 1 + naulas,  
y + 1).Value  
    Next y  
Next x  
End If  
  
End Sub
```


Anexo IV. Código de la función Ordenar_Matriz

```
Sub Ordenar_Matriz(ByRef matrizOr() As Long, columna As Integer)

Dim aux As Long

    If UBound(matrizOr, 2) < Col Then
        MsgBox "La matriz no tiene suficientes columnas para ordenar por la
columna especificada.", vbExclamation
        Exit Sub
    End If

    'Algoritmo de ordenamiento (burbuja)
    For i = LBound(matrizOr, 1) + 1 To UBound(matrizOr, 1) - 1
        For j = i + 1 To UBound(matrizOr, 1)
            If matrizOr(i, columna) < matrizOr(j, columna) Then
                For k = LBound(matrizOr, 2) + 1 To UBound(matrizOr, 2)
                    aux = matrizOr(i, k)
                    matrizOr(i, k) = matrizOr(j, k)
                    matrizOr(j, k) = aux
                Next k
            End If
        Next j
    Next i

End Sub
```

Anexo V. Código de la heurística de búsqueda local

```
Private Sub CommandButton2_Click()  
  
Dim aux As Integer  
  
Randomize  
  
Dim Au As Integer 'número de aulas  
Dim dia As Integer  
Dim CodClas As Integer  
Dim Curso As Integer  
Dim OcLib As Integer  
Dim Aula As Integer  
Dim Carrera As Integer  
Dim Cuatri As Integer  
Dim Equiv As Integer  
DimCodigo_Asig As Integer  
Dim Curso_Asig As Integer  
Dim Carrera_Asig As Integer  
  
Dim Asi As Integer 'número de asignaturas  
Dim AsEq As Integer 'número de parejas en listado asignaturas equivalentes  
Dim AuPar As Integer 'número de parejas de aulas  
Dim Aux1 As Integer  
Dim NOpt As Integer  
Dim PosOpt As Integer  
  
Dim sheetWork As String  
Dim Col As Integer  
Dim fila As Integer  
Dim FilaCod As Integer  
  
Dim Datos_Asignaturas() As Long  
Dim Cod_Asig_Eq() As Long
```

```

Dim Datos_Capacidad() As Long
Dim Datos_Distancias() As Long
Dim Disponibilidad() As Integer

Dim Horario_Solucion() As String

Dim Compatibilidad() As Long 'Esta matriz va a tener 3 dimensiones: titulación,
curso y día. De manera que tendrá cuatro columnas.
'La 1º alberga la titulación, la 2º el curso, la 3º el día y la 4º un valor
binario
'(0 si ese curso y titulación aun no tiene asignado examen ese día, o 1 si si
que tiene asignado un examen ese día).

'Creo una matriz que sea asignada donde se marque para cada asignatura si con 1
si ya se ha puesto fecha o no se tiene que poner fecha, y con 0 si falta por
poner fecha.
Dim asignada() As Long

Dim nombre As String
Dim AsigEq() As Long

Dim Optativas() As Long

.....

'Lectura de datos directos
.....

'Hay tres datos que se le tienen que proporcionar a la heurística
'De manera que el usuario lo escriba en la Excel antes de ejecutar la heurística
'Estos datos son: el número de días, el cuatrimestre, y el turno.
Dim numdias As Integer 'número de días en los que se puede programar examen
Dim cuatrimestre As Integer 'indica el turno que se quiere programar (1: A, 2:
B)
Dim turno As Integer 'indica el turno que se quiere programar (1: ma-ana, 2:
tarde, 0: ambos)

numdias = Worksheets("HEURÍSTICA").Cells(7, 3).Value
cuatrimestre = Worksheets("HEURÍSTICA").Cells(5, 3).Value
turno = Worksheets("HEURÍSTICA").Cells(6, 3).Value

.....

```

```
'Leer tablas de datos
.....

Call Calcular_Tamaño("DATOS CAPACIDAD", Au, Col)
Call Leer_Tabla_Col("DATOS CAPACIDAD", Au, Col, Datos_Capacidad())

'Esta misma estructura se puede emplear ahora para leer el resto de datos que
disponen de tablas con los datos en columnas

'Que serán los que están alojados en la pestaña: "DATOS ASIGNATURAS COD", "CÓD.
ASIG. EQ" y "DATOS DISTANCIA"

Call Calcular_Tamaño("DATOS DISTANCIAS", AuPar, Col)
Call Leer_Tabla_Col("DATOS DISTANCIAS", AuPar, Col, Datos_Distancias())

Ndis = AuPar

Call Calcular_Tamaño("DATOS ASIGNATURAS COD", Asi, Col)
Call Leer_Tabla_Col("DATOS ASIGNATURAS COD", Asi, Col, Datos_Asignaturas())

Call Calcular_Tamaño("CÓD. ASIG. EQ.", AsEq, Col)
Call Leer_Tabla_Col("CÓD. ASIG. EQ.", AsEq, Col, Cod_Asig_Eq())
ReDim AsigEq(AsEq)

Call Calcular_Tamaño("COMPATIBILIDAD", Aux1, Col)
Call Leer_Tabla_Col("COMPATIBILIDAD", Aux1, Col, Compatibilidad())

Call Calcular_Tamaño("OPTATIVAS", NOpt, Col)
Call Leer_Tabla_Col("OPTATIVAS", NOpt, Col, Optativas())

'Por otra parte habría que definir una función diferente para la lectura de
tablas que tienen rotulos tanto en la primera fila como en la primera columna
'Entiendo que es la función que tienes puesta como Leer_Disponibilidad.
Call Leer_Disponibilidad("DATOS DISPONIBILIDAD", Au, turno, numdias + 1,
Disponibilidad())

.....

' CREACION DE LA MATRIZ SOLUCION
.....

If turno = 1 Or turno = 2 Then
    ReDim Horario_Solucion(Au, numdias)
ElseIf turno = 0 Then
```

```

ReDim Horario_Solucion(2 * Au, numdias)
End If

.....

'Ordenamos las clases en función de su capacidad
.....

Call Ordenar_Matriz(Datos_Capacidad(), 3)

Call Ordenar_Matriz(Datos_Asignaturas(), 5)

'Actualizamos el vector de asignada() para que solamente se tengan en cuenta las
asignaturas del turno y cuatrimestre que tocan:

'En la primera columna pongo el número de la asignatura y en la segunda el valor
binario
ReDim asignada(Asi, 2)
For x = 1 To Asi
    asignada(x, 1) = Datos_Asignaturas(x, 4)
    If (Datos_Asignaturas(x, 3) = cuatrimestre Or Datos_Asignaturas(x, 3) = 0)
Then 'Comprobamos que la asignatura se imparte en el cuatrimestre que vamos a
programar o es anual.
        If turno = 1 Then 'Si es turno de mañana comprobamos que sea asignatura
de 1º o 3º, en caso contrario le ponemos que ya está asignada para que no la
tenga en cuenta.
            If (Datos_Asignaturas(x, 2) = 1 Or Datos_Asignaturas(x, 2) = 3) Then
                asignada(x, 2) = 0
            ElseIf (Datos_Asignaturas(x, 2) = 2 Or Datos_Asignaturas(x, 2) = 4)
Then
                asignada(x, 2) = 1
            End If
        ElseIf turno = 2 Then 'Lo contrario para cuando sea el turno de tarde
            If (Datos_Asignaturas(x, 2) = 1 Or Datos_Asignaturas(x, 2) = 3) Then
                asignada(x, 2) = 1
            ElseIf (Datos_Asignaturas(x, 2) = 2 Or Datos_Asignaturas(x, 2) = 4)
Then
                asignada(x, 2) = 0
            End If
        ElseIf turno = 0 Then 'Si se desea programar todos los turnos (mañana y
tarde) entonces todas las asignaturas del cuatri se deben poner como pendientes
de asignar.

```

```

    asignada(x, 2) = 0
End If

Else 'En caso contrario, la asignatura se imparte en un cuatri que no
queremos programar y por tanto, no se debe asignar
    asignada(x, 2) = 1
End If
Next x

.....

' Asignación clases y horario
.....

'Necesito para hacer más fácil la asignación un matriz (turno, dias) donde se
guarde por cada día la capacidadtotal disponible teniendo en cuenta las aulas
que no se han utilizado aún
Dim capacidadtotal() As Integer
ReDim capacidadtotal(2, numdias)
For x = 1 To Au
    For y = 1 To numdias
        capacidadtotal(1, y) = capacidadtotal(1, y) + Datos_Capacidad(x, 3)
        capacidadtotal(2, y) = capacidadtotal(2, y) + Datos_Capacidad(x, 3)
    Next y
Next x

'Lo primero que hay que hacer es comprobar de asignatura con mayor a menor
matriculados si cabe en el aula con mayor a menor capacidad
'Hay que coger la asignatura con más alumnos y ver si hay que programar su examen
o no:
For x = 1 To Asi
    Asignatura = Datos_Asignaturas(x, 4)

    If asignada(x, 2) = 0 Then 'Si la asignatura está por asignar

        .....

        ''' IDENTIFICAR TURNO ASIGNATURA '''
        .....

        'Identificar el turno en que se debe hacer la asignatura. Si es de 1 o
3 curso, por la ma-ana. Si es de 2 o 4 curso, por la tarde.

        'A no ser que sea optativa, que en ese caso, se haría en el turno opuesto

```

```

PosOpt = 0
If Datos_Asignaturas(x, 2) = 1 Or Datos_Asignaturas(x, 2) = 3 Or
Datos_Asignaturas(x, 2) = 5 Then
    TurnoAs = 1
    For y = 1 To NOpt
        If Asignatura = Optativas(y, 1) Then
            TurnoAs = 2
            PosOpt = y
            y = NOpt
        End If
    Next y
ElseIf Datos_Asignaturas(x, 2) = 2 Or Datos_Asignaturas(x, 2) = 4 Then
    TurnoAs = 2
    For y = 1 To NOpt
        If Asignatura = Optativas(y, 1) Then
            TurnoAs = 1
            PosOpt = y
            y = NOpt
        End If
    Next y
End If

```

```

.....
''' COMPROBAR SI ASIGNATURA TIENE EQUIVALENTES '''
.....

```

```

ContadorAsEq = 0
MatriculadosEq = Datos_Asignaturas(x, 5)
nombre = Asignatura

For i = 1 To AsEq
    If Cod_Asig_Eq(i, 1) = Asignatura Then
        AsignaturaEq = Cod_Asig_Eq(i, 2)
        ContadorAsEq = ContadorAsEq + 1
        AsigEq(ContadorAsEq) = Cod_Asig_Eq(i, 2)
        nombre = nombre & ";" & AsignaturaEq 'El nombre que tenga hasta
el momento le añade ";" y el nombre de la nueva asignatura equivalente
        For j = 1 To Asi
            If AsignaturaEq = Datos_Asignaturas(j, 4) Then

```



```

5)           MatriculadosEq = MatriculadosEq + Datos_Asignaturas(j,
           j = Asi
           End If
           Next j
           End If
Next i
For j = ContadorAsEq + 1 To AsEq
  AsigEq(j) = 0
Next j

'Al acabar este bucle, en matal está el número de matriculados, en
ContadorAsEq el número de asignaturas equivalentes,

'y en AsigEq() el listado de asignaturas equivalentes a la asignatura
Asignatura

.....
''''
''' COMPROBAR EL PRIMER DIA EN QUE SE PUEDE PROGRAMAR ASIGNATURA Y
EQUIVALENTES '''
.....
''''

dia = 0
capacidadmax = 0
For t = 1 To numdias
  compatible = 0

  If PosOpt = 0 Then ' Si la asignatura es normal (es decir, no es una
  optativa). Esto se comprueba porque en el caso de optativas, no tienen
  imcompatibilidades (se pueden hacer varias optativas el mismo día ya que el
  alumno usualmente escoge una).

    'Primero miro si la "Asignatura" se puede hacer en ese día
    For i = 1 To Aux1
      If Compatibilidad(i, 1) = Datos_Asignaturas(x, 1) And
      Compatibilidad(i, 2) = Datos_Asignaturas(x, 2) And Compatibilidad(i, 3) = t And
      Compatibilidad(i, 5) = Datos_Asignaturas(x, 6) Then
        compatible = compatible + Compatibilidad(i, 4)
        i = Aux1
      End If
    Next i

    'Luego compruebo si sus equivalentes se pueden hacer en ese día
    If ContadorAsEq > 0 Then

```

```

    For y = 1 To ContadorAsEq
        For k = 1 To Asi 'Localizo en que fila de
Datos_Asignaturas está la asignatura equivalente (PosEq)
            If Datos_Asignaturas(k, 4) = AsigEq(y) Then
                PosEq = k
                k = Asi
            End If
        Next k
        For i = 1 To Aux1
            If Compatibilidad(i, 1) = Datos_Asignaturas(PosEq,
1) And Compatibilidad(i, 2) = Datos_Asignaturas(PosEq, 2) And Compatibilidad(i,
3) = t And Compatibilidad(i, 5) = Datos_Asignaturas(PosEq, 6) Then
                compatible = compatible + Compatibilidad(i, 4)
                i = Aux1
            End If
        Next i
    Next y
End If

End If

'En caso de que en este punto el contador de compatible sea igual a
0 querría decir que todas las asignaturas equivalentes también son compatibles,
'por lo que nos guardamos ese día para hacer la programación.
'Siempre y cuando, quepan los matriculados en la asignatura y
equivalentes
'Le voy a añadir que compruebe si es el día con mayor capacidad
disponible, así intentamos que no se llenen los primeros días y se arregle el
problema de que hay
'asignaturas sin ningún día compatible.

If compatible = 0 And capacidadtotal(TurnoAs, t) >= MatriculadosEq
And capacidadtotal(TurnoAs, t) > capacidadmax Then

    If capacidadtotal(TurnoAs, t) > 0 Then 'Esto ha sido
neesario ponerlo porque sino quería asignar asignatura sin matriculados a día
que no tiene aulas libres. y igualmente hay que asignar aula...

        día = t
        't = numdías
        capacidadmax = capacidadtotal(TurnoAs, t)
    End If

End If

Next t

'Al acabar esto tenemos guardado en día el día en el que queremos
programar

```

'Si aquí día fuera 0, tendríamos un problema. Querría decir que no ha encontrado ningún día en que poder hacer la asignatura y sus equivalentes.

'Esperemos que esto no ocurra.

```

.....
''' ASIGNACIÓN DE AULA '''
.....

For y = 1 To Au
  Aula = Datos_Capacidad(y, 1)
  'Identificar la posición del aula en la matriz disponibilidad
  For j = 1 To Au
    If Aula = Disponibilidad(j, 1) Then
      Posaula = j
      j = Au
    End If
    If TurnoAs = 2 Then
      Posaula = Posaula + Au
    End If
  Next j

  'Comprobar que el aula está disponible en el día
  If Disponibilidad(Posaula, dia + 1) = 0 Then
    MatriculadosEqaux = MatriculadosEq 'Creo esta variable para
    poder ir actualizando el valor y no perder lo de MatriculadosEq

    'Comprobar si caben los alumnos en una sola aula, y asignarsela
    If MatriculadosEq <= Datos_Capacidad(y, 3) Then

      'Antes de asignarle el aula, miro si hay otra más pequeña
      que le pueda venir bien, para dejar libres las grandes
      mejorcap = Datos_Capacidad(y, 3)
      PosaulaC = y
      If TurnoAs = 1 Then
        idt = 0
      ElseIf TurnoAs = 2 Then
        idt = Au
      End If
      For i = y + 1 To Au

```

```

        For j = 1 + idt To Au + idt
            If Datos_Capacidad(i, 1) = Disponibilidad(j, 1) And
Disponibilidad(j, dia + 1) = 0 Then
                If MatriculadosEq <= Datos_Capacidad(i, 3) And
mejorcap > Datos_Capacidad(i, 3) Then
                    PosaulaC = i
                    Posaula = j
                    mejorcap = Datos_Capacidad(i, 3)
                End If
            End If
        Next j
    Next i

```

```

        Horario_Solucion(Posaula, dia) = nombre
        Disponibilidad(Posaula, dia + 1) = 1
        capacidadtotal(TurnoAs, dia) = capacidadtotal(TurnoAs, dia)
- Datos_Capacidad(PosaulaC, 3)
        MatriculadosEqaux = MatriculadosEqaux -
Datos_Capacidad(PosaulaC, 3)

```

```

        asignada(x, 2) = 1 'La asignatura x ya se ha programado
        If ContadorAsEq > 0 Then 'Bucle para asignar las
equivalentes (decir que ya se ha programado su examen)
            For i = 1 To ContadorAsEq
                For j = 1 To Asi
                    If Datos_Asignaturas(j, 4) = AsigEq(i) Then
                        asignada(j, 2) = 1
                        j = Asi
                    End If
                Next j
            Next i
        End If

```

```

        y = Au 'Para que salga del "For y" y se vaya a la siguiente
asignatura

```

'En el caso en que no quepan, hay que asignar el aula inicial que hemos seleccionado ya, y también las aulas más cercanas hasta cubrir los alumnos

```

ElseIf MatriculadosEq > Datos_Capacidad(y, 3) Then
    Horario_Solucion(Posaula, dia) = nombre
    Disponibilidad(Posaula, dia + 1) = 1
    capacidadtotal(TurnoAs, dia) = capacidadtotal(TurnoAs, dia)
- Datos_Capacidad(y, 3)
    MatriculadosEqaux = MatriculadosEq - Datos_Capacidad(y, 3)

    asignada(x, 2) = 1
    If ContadorAsEq > 0 Then
        For i = 1 To ContadorAsEq
            For j = 1 To Asi
                If Datos_Asignaturas(j, 4) = AsigEq(i) Then
                    asignada(j, 2) = 1
                    j = Asi
                End If
            Next j
        Next i
    End If

    'Hasta aquí es igual que el caso en que cabe en un aula.
    ahora vamos a ver que hacer con el resto de alumnos:

    'AAhora vamos a proceder a asignar otras aulas, hasta que
    se complete el número de matriculados
    Do While MatriculadosEqaux > 0
        min = 10000
        For i = 1 To AuPar
            If Datos_Distancias(i, 1) = Aula Then
                Aula2 = Datos_Distancias(i, 2)
                'Localizamos en que fila de la tabla
disponibilidad está el aula2
                For k = 1 To Au
                    If Aula2 = Disponibilidad(k, 1) Then
                        If TurnoAs = 1 Then
                            PosaulaD = k
                            k = Au
                        ElseIf TurnoAs = 2 Then
                            PosaulaD = k + Au
                            k = Au
                        End If
                    End If
                End If
            End If
        End If
    End If

```

```

Next k
'Localizamos en que fila de la tabla capacidades
está el aula2

For k = 1 To Au
    If Aula2 = Datos_Capacidad(k, 1) Then
        PosaulaC = k
        k = Au
    End If
Next k

'En el caso que tenga menor distancia, y este
disponible, se guarda el nuevo valor mínimo de distancia y el aula (en mejor)
If Disponibilidad(PosaulaD, dia + 1) = 0 And
Datos_Distancias(i, 3) < min Then
    min = Datos_Distancias(i, 3)
    PosaulaDmejor = PosaulaD
    PosaulaCmejor = PosaulaC
    aulamejor = Aula2

ElseIf Disponibilidad(PosaulaD, dia + 1) = 0 And
Datos_Distancias(i, 3) = min Then
    'En el caso de empate por distancia entre
Aula2 (nueva aula candidata) y aulamejor (mejor aula candidata hasta el momento)
    'Se escoge la que mejor se ajuste a los
alumnos que quedan por ubicar

    If Abs(Datos_Capacidad(PosaulaC, 3) -
MatriculadosEqaux) < Abs(Datos_Capacidad(PosaulaCmejor, 3) - MatriculadosEqaux)
Then
        min = Datos_Distancias(x, 3)
        PosaulaDmejor = PosaulaD
        PosaulaCmejor = PosaulaC
        aulamejor = Aula2
    End If
End If
End If
Next i

'Como resultado de este for tendré un aula seleccionada
para completar la solución.

'Ahora actualizo la solución con esta aula
Horario_Solucion(PosaulaDmejor, dia) = nombre
Disponibilidad(PosaulaDmejor, dia + 1) = 1
MatriculadosEqaux = MatriculadosEqaux -
Datos_Capacidad(PosaulaCmejor, 3)

```

```

        capacidadtotal(TurnoAs, dia) = capacidadtotal(TurnoAs,
dia) - Datos_Capacidad(PosaulaCmejor, 3)
        Loop
        y = Au
    End If
End If
Next y

.....
''' ACTUALIZAR COMPATIBILIDAD DE LAS TITULACIONES Y CURSOS '''
.....

'Solo hay que hacer esto de actualizar la compatibilidad, si las
asignaturas no son optativas.

'Esto se debe a que si son optativas se pueden solapar sin problema.
If PosOpt = 0 Then
    If ContadorAsEq = 0 Then
        For i = 1 To Aux1
            If Compatibilidad(i, 1) = Datos_Asignaturas(x, 1) And
Compatibilidad(i, 2) = Datos_Asignaturas(x, 2) And Compatibilidad(i, 3) = dia
And Compatibilidad(i, 5) = Datos_Asignaturas(x, 6) Then
                Compatibilidad(i, 4) = 1
                i = Aux1
            End If
        Next i
    End If
    If ContadorAsEq > 0 Then
        For i = 1 To Aux1
            If Compatibilidad(i, 1) = Datos_Asignaturas(x, 1) And
Compatibilidad(i, 2) = Datos_Asignaturas(x, 2) And Compatibilidad(i, 3) = dia
And Compatibilidad(i, 5) = Datos_Asignaturas(x, 6) Then
                Compatibilidad(i, 4) = 1
                i = Aux1
            End If
        Next i
        For y = 1 To ContadorAsEq
            For k = 1 To Asi 'Localizo en que fila de Datos_Asignaturas
está la asignatura equivalente (PosEq)
                If Datos_Asignaturas(k, 4) = AsigEq(y) Then
                    PosEq = k
                    k = Asi
                End If
            Next k
        Next y
    End If
End If

```

```

        End If
    Next k
    For i = 1 To Aux1
        If Compatibilidad(i, 1) = Datos_Asignaturas(PosEq, 1)
        And Compatibilidad(i, 2) = Datos_Asignaturas(PosEq, 2) And Compatibilidad(i, 3)
        = dia And Compatibilidad(i, 5) = Datos_Asignaturas(PosEq, 6) Then
            Compatibilidad(i, 4) = 1
            i = Aux1
        End If
    Next i
    Next y
    End If
End If

End If

Next x

'IMPRESIÓN SOLUCIÓN CONSTRUCTIVA
For x = 1 To 2 * Au
    For y = 1 To numdias
        Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(x + 2, y + 17).Value =
Horario_Solucion(x, y)
    Next y
Next x

.....
''' EVALUACIÓN SOLUCIÓN '''
.....

distanciatotal = 0
If turno = 0 Then
    nfil = 2 * Au
Else
    nfil = Au
End If
For t = 1 To numdias
    For i = 1 To nfil - 1
        For j = i + 1 To nfil
            If Horario_Solucion(i, t) <> "" Then

```


If Horario_Solucion(i, t) = Horario_Solucion(j, t) Then 'Si coincide la asignatura programada en dos aulas hay que tener en cuenta su distancia.

```

        Aula1 = Disponibilidad(i, 1)
        Aula2 = Disponibilidad(j, 1)
        For x = 1 To AuPar
            If Aula1 = Datos_Distancias(x, 1) And Aula2 =
Datos_Distancias(x, 2) Then
                distanciatotal = distanciatotal +
Datos_Distancias(x, 3)
            End If
        Next x
    End If
End If
Next j
Next i

```

```

Next t
Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(3, 29).Value = distanciatotal

```

```

holguratotalsolucion = 0

```

```

capacidadtotalsolucion = 0

```

```

matriculadostotalsolucion = 0

```

```

If turno = 0 Then

```

```

    numturnos = 2

```

```

Else

```

```

    numturnos = 1

```

```

End If

```

```

For x = 1 To Au

```

```

    For y = 1 To Au * numturnos

```

```

        If Datos_Capacidad(x, 1) = Disponibilidad(y, 1) Then

```

```

            For t = 1 To numdias

```

```

                capacidadtotalsolucion = capacidadtotalsolucion +
Datos_Capacidad(x, 3) * Disponibilidad(y, t + 1)

```

```

            Next t

```

```

        End If

```

```

    Next y

```

```

Next x

```

```

For x = 1 To Asi

```

```

    If cuatrimestre = Datos_Asignaturas(x, 3) Then

```

```

        matriculadostotalsolucion = matriculadostotalsolucion +
Datos_Asignaturas(x, 5)

```

```
End If
Next x
holguratotalsolucion = capacidadtotalsolucion - matriculadostotalsolucion
Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(6, 29).Value = holguratotalsolucion

'ALGORITMO DE BÚSQUEDA LOCAL

'Voy a poner todas las definiciones juntas para tener un código más compacto.
Dim diaal As Integer
Dim turnoal As Integer
Dim aulaal As Integer 'No te funcionaba porque las aulas son números, no
palabras. Te lo he corregido a "Integer" en vez de String
Dim Filaal As Integer

Dim nombreal As String
Dim asigalS() As String 'He cambiado el nombre a esta para ser coherente con la
siguiente.
Dim asigalL() As Long 'He creado esta para almacenar las asignaturas equivalentes
en un vector Long
Dim num As Long

Dim matal As Long
Dim cursoal As Integer
Dim tital() As Long
Dim numasig As Integer
Dim Posasigal() As Long

Dim AsLong As Integer
Dim nombrealL As Long

Dim diadest As Integer
Dim PosCompEq() As Integer

Dim aulaorigen() As Integer
Dim auladestino() As Integer

Dim distin As Long
```

```
Dim distfin As Long
```

```
For iteración = 1 To 10
```

```

.....
''' SELECCIÓN ASIGNATURA ORIGEN '''
.....

'Hay que repetir todo el proceso de elegir aleatoriamente día, turno y
aula hasta que encuentre una combinación en la que haya guardado algo...

'Es decir, hay que repetir este bucle hasta que encuentre un día, turno
y aula en el que haya alguna asignatura.

Do
    'Escoger un día aleatorio entre 1 y 10
    diaal = GenerarAleatorio(1, numdias)
    'Escoger un turno al azar
    turnoal = GenerarAleatorio(1, 2)
    'Escoger un aula al azar. Va a hacer referencia a la fila de la
solución
    aulaal = GenerarAleatorio(1, Au)
    'Extraemos la asignatura
    If turnoal = 1 Then
        nombreal = Horario_Solucion(aulaal, diaal)
        idt = 0
    Else
        nombreal = Horario_Solucion(aulaal + Au, diaal)
        idt = Au
    End If
    If nombreal <> "" Then
solucionada
        'Registramos las aulas en las que está la asignatura
        ReDim aulaorigen(Au)
        For y = 1 To Au
            'si el aula que está en la fila y tiene guardada las
asignaturas que queremos mover, ponemos un 1 en su posición, y sino un 0 en un
vector binario que he creado.
            If turnoal = 1 Then
                If Horario_Solucion(y, diaal) = nombreal Then
                    aulaorigen(y) = 1
                Else
                    aulaorigen(y) = 0
            End If
        Next y
    End If
End Do

```

```

        End If
    ElseIf turnoal = 2 Then
        If Horario_Solucion(y + Au, diaal) = nombreal Then
            aulaorigen(y) = 1
        Else
            aulaorigen(y) = 0
        End If
    End If
Next y

'Calculamos el número de aulas asignadas inicialmente a esa
asignatura
inicio = 0
For y = 1 To Au
    If aulaorigen(y) = 1 Then
        inicio = inicio + 1
    End If
Next y
End If

'Todo esto se repite mientras la celda aleatoria esté vacía o la
asignatura solo tenga asignada una aula (no tiene sentido cambiarla entonces)
Loop While Horario_Solucion(aulaal + idt, diaal) = "" 'Or inicio = 1

.....
''' IDENTIFICAR ASIGNATURAS Y EQUIVALENTES NOMBREAL '''
.....

'Identificar si es asignatura única o tiene equivalentes
AsLong = Len(nombreal)

'Esto es para saber el número de caracteres que tiene nombreal guardados.
si es 5 solo hay una asignatura,
'si hay más, hay que hacer el split para sacar las equivaentes (NEq).
If AsLong = 5 Then
    nombrealL = CLng(nombreal)
    NEq = 1
Else
    'Redimensiono aquí los vectores para que se vacíen
    ReDim asigalS(AsEq)
    ReDim asigalL(AsEq)

```

```

asigalS() = Split(nombreal, ";")
y = 0
aux = UBound(asigalS)
Do While y <= UBound(asigalS)
    asigalL(y) = CLng(asigalS(y))
    y = y + 1
Loop
NEq = y
End If

.....
''' IDENTIFICAR NÚMERO MATRÍCULADOS '''
.....

matal = 0
For y = 1 To Asi
    If NEq = 1 Then 'si no tiene equivalentes, es decir, solo es una
asignatura
        If nombrealL = Datos_Asignaturas(y, 4) Then 'Localiza la
asignatura en la matriz de datos
            matal = Datos_Asignaturas(y, 5)
        End If
    ElseIf NEq > 1 Then
        For x = 0 To NEq - 1
            If asigalL(x) = Datos_Asignaturas(y, 4) Then
                matal = matal + Datos_Asignaturas(y, 5)
            End If
        Next x
    End If
Next y

.....
''' IDENTIFICAR SI ES OPTATIVA '''
.....

opt = 0
For y = 1 To NOpt
    If nombrealL = Optativas(y, 1) Then
        opt = opt + 1
    End If

```

```

If NEq > 1 Then
  For x = 0 To NEq - 1
    If asigalL(x) = Optativas(y, 1) Then
      opt = opt + 1
    End If
  Next x
End If

Next y

'Al acabar esto, en opt tendremos guardado 0 si no es optativa, o un
valor superior a 0 si alguna de las asignaturas es optativa.

'Identificar el turno en que se debe hacer la asignatura. Si es de 1 o
3 curso, por la ma-ana. Si es de 2 o 4 curso, por la tarde.

'A no ser que sea optativa, que en ese caso, se haría en el turno opuesto

.....
''' SELECCIÓN DIA DESTINO '''
.....

'Voy a hacer que repita el proceso 5 veces, para encontrar un día en el
que quepa y sea compatible... aux = 1 significa que SI se ha encontrado día.

aux = 0
'diadestdef=0
For iter = 1 To 5
  diadest = GenerarAleatorio(1, numdias)

  'Una vez identificado el día de destino, hay que ver si ya hay otra
asignatura de ese curso y titulación en ese día (esto si no es optativa).

  compdest = 0

  If opt = 0 Then 'si no es optativa
    If NEq = 1 Then 'Si no tiene equivalentes, es decir, solo es una
asignatura.

      For y = 1 To Asi
        If nombrealL = Datos_Asignaturas(y, 4) Then
          For x = 1 To Aux1
            If Compatibilidad(x, 1) = Datos_Asignaturas(y,
1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And Compatibilidad(x, 3)
= diadest And Compatibilidad(x, 5) = Datos_Asignaturas(y, 6) Then
              compdest = Compatibilidad(x, 4) 'pondrá 0
si es compatible, 1 si no lo es

              x = Aux1
            End If
          Next x
        End If
      Next y
    End If
  Next iter
End If

```

```

Next y
ElseIf NEq > 1 Then 'Si que tiene asignaturas equivalentes.
  For j = 0 To NEq - 1
    For y = 1 To Asi
      If asigalL(j) = Datos_Asignaturas(y, 4) Then
        For x = 1 To Aux1
          If Compatibilidad(x, 1) =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diadest And Compatibilidad(x, 5) = Datos_Asignaturas(y,
6) Then
            compdest = compdest + Compatibilidad(x,
4)
            x = Aux1
          End If
        Next x
        i = Asi
      End If
    Next y
  Next j
End If
End If

'Comprobar si el día es válido (por capacidad y compatibilidad y por
ser día diferente)
If diadest <> diaal And capacidadtotal(turnoal, diadest) >= matal
And compdest = 0 Then
  aux = 1
  diadestdef = diadest
End If
Next iter
diadest = diadestdef

.....
''' SELECCIÓN AULAS DÍA DESTINO '''
.....

'Si aux es 1 quiere decir que si puede hacer el intercambio por razones
de capacidad y compatibilidad,
'y entonces entramos en la generación de vecino, sino pasamos a la
siguiente iteración.
'Aquí generamos la solución vecina
If aux = 1 Then

```

```
ReDim auladestino(Au) 'Para vaciarlo cada vez, y partir de 0

.....

''' CASO DE TENER INICIALMENTE 1 AULA '''
.....

'En el caso en que la asignatura/asignaturas seleccionada/s solo
tenga asignada una aula en inicialmente

'Interesa que evalúe otra aula que esté más ajustada en capacidad
'de esta manera liberaría aulas más grandes para poner otras
asignaturas

If inicio = 1 Then 'Si solo tenía un aula
    'Vamos a comprobar qué aula es y ver la holgura que tiene
    For y = 1 To Au 'Este bucle es para saber en que fila de
Datos_Capacidad está guardada el aula
        If aulaorigen(y) = 1 Then
            For x = 1 To Au
                If Disponibilidad(y, 1) = Datos_Capacidad(x, 1) Then
                    Holgurainicial = Datos_Capacidad(x, 3) - matal
                    capacidadaulaorigen = Datos_Capacidad(x, 3)
                    x = Au
                    y = Au
                End If
            Next x
        End If
    Next y

    'Ahora vamos a buscar el aula del diadest que más se ajuste a
los matriculados

    Holgurafinal = 10000 'Ponemos un valor grande para que luego
compare y lo reemplace

    For y = 1 To Au
        If turnoal = 1 Then
            k = y
        ElseIf turnoal = 2 Then
            k = y + Au
        End If

        If Disponibilidad(k, diadest + 1) = 0 Then 'Compruebo si el
aula está disponible

            For x = 1 To Au
                If Disponibilidad(y, 1) = Datos_Capacidad(x, 1) Then
'Busco el aula en la matriz de capacidades
```



```

                                If Datos_Capacidad(x, 3) >= matal Then 'Si la
asignatura cabe en ese aula
                                If Datos_Capacidad(x, 3) - matal <
Holgurafinal Then 'si es mejor en cuanto a algura que las que se han analizado
hasta ahora, se guarda
                                Posaula = y 'En aux tengo guardada el
aula que selecciono
                                Holgurafinal = Datos_Capacidad(x, 3) -
matal
                                capacidadauladestino
                                =
Datos_Capacidad(x, 3)
                                End If
                                End If
                                End If
                                Next x
                                End If
                                Next y

.....
''' COMRPOBAR SI MEJORA LA SOLUCIÓN '''
.....
'Solo falta comparar la holgura inicial con la final para ver si
mejora.
                                If Holgurafinal < Holgurainicial Then
                                auladestino(Posaula) = 1

.....
''' BORRAR SOLUCIÓN DÍA ORIGEN '''
.....
'Solución
                                'Y en el caso de mejorar por último faltaría actualizar la
solución
                                'Primero lo borro del día de origen
                                For y = 1 To Au
                                If turnoal = 1 Then
                                k = y
                                ElseIf turnoal = 2 Then
                                k = y + Au
                                End If
                                If aulaorigen(y) = 1 Then

```

```

        Disponibilidad(k, diaal + 1) = 0 'El aula origen
vuelve a estar disponible
        Horario_Solucion(k, diaal) = "" 'En horario solución
se vacía la casilla
    End If
Next y
    'La capacidad total del día y turno se incrementa en la
capacidad del aula que hemos liberado
    capacidadtotal(turnoal, diaal) = capacidadtotal(turnoal,
diaal) + capacidadaulaorigen
    'Ese día ya estaría disponible para hacer cualquier examen
de ese título y curso. es decir, actualizar compatibilidad (solo si no es
optativa).
    If opt = 0 Then
        If NEq = 1 Then 'Si no tiene equivalentes, es decir,
solo es una asignatura.
            For y = 1 To Asi
                If nombrealL = Datos_Asignaturas(y, 4) Then
                    For x = 1 To Aux1
                        If Compatibilidad(x, 1) =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diaal And Compatibilidad(x, 5) = Datos_Asignaturas(y, 6)
Then
                            Compatibilidad(x, 4) = 0 'Se pone
que ya es compatible, porque hemos quitado la asignatura de este día.
                            x = Aux1
                        End If
                    Next x
                End If
            Next y
        Else 'Si que tiene asignaturas equivalentes.
            For j = 0 To NEq - 1
                For y = 1 To Asi
                    If asigall(j) = Datos_Asignaturas(y, 4)
Then
                        For x = 1 To Aux1
                            If Compatibilidad(x, 1) =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diaal And Compatibilidad(x, 5) = Datos_Asignaturas(y, 6)
Then
                                    Compatibilidad(x, 4) = 0
                                    x = Aux1
                                End If
                            Next x
                        y = Asi
                    End If
                End If
            Next j
        End If
    End If

```

```

        End If
    Next y
Next j
End If
End If

.....
''' ESCRIBIR SOLUCIÓN DÍA DESTINO '''
.....

'Y en el caso de mejorar por último faltaría actualizar la
solución

'Primero lo borro del día de origen
For y = 1 To Au
    If turnoal = 1 Then
        k = y
    ElseIf turnoal = 2 Then
        k = y + Au
    End If
    If auladestino(y) = 1 Then
        Disponibilidad(k, diadest + 1) = 1 'El aula destino
ya no está disponible
        Horario_Solucion(k, diadest) = nombreal 'En horario
solución se pone el nombreal
    End If
Next y
'La capacidad total del día y turno se incrementa en la
capacidad del aula que hemos liberado
    capacidadtotal(turnoal, diadest) = capacidadtotal(turnoal,
diadest) - capacidadauladestino
'Ese día ya NO estaría disponible para hacer cualquier
examen de ese título y curso. es decir, actualizar compatibilidad
'Solo para las no optativas
If opt = 0 Then
    If NEq = 1 Then 'Si no tiene equivalentes, es decir,
solo es una asignatura.
        For y = 1 To Asi
            If nombrealL = Datos_Asignaturas(y, 4) Then
                For x = 1 To Aux1
                    If      Compatibilidad(x,      1)      =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diadest And Compatibilidad(x, 5) = Datos_Asignaturas(y,
6) Then

```

```

                                Compatibilidad(x, 4) = 1 'Se pone
que ya NO es compatible

                                x = Aux1
                                End If
                                Next x
                                End If
                                Next y
                                Else 'Si que tiene asignaturas equivalentes.
                                For j = 0 To NEq - 1
                                For y = 1 To Asi
                                If asigalL(j) = Datos_Asignaturas(y, 4)
Then
                                For x = 1 To Aux1
                                If Compatibilidad(x, 1) =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diadest And Compatibilidad(x, 5) = Datos_Asignaturas(y,
6) Then
                                Compatibilidad(x, 4) = 1
                                x = Aux1
                                End If
                                Next x
                                y = Asi
                                End If
                                Next y
                                Next j
                                End If
                                End If
                                End If
                                End If

                                .....
                                ''' CASO DE TENER INICIALMENTE MÁS DE 1 AULA '''
                                .....

                                'En caso de tener más de un aula asignada inicialmente a la
asignatura
                                'vamos a proceder como en la heurística constructiva, pero teniendo
en cuenta el día de destino y viendo las aulas que quedan allí disponibles.
                                'Ya hemos comprobado antes que en el día destino, esta asignatura y
sus equivalentes habían. por lo que no hay que comprobarlo de nuevo
                                If inicio > 1 Then

```

For y = 1 To Au 'Vamos a recorrer la matriz de capacidad, para ir cogiendo las aulas de mayor a menor capacidad.

Aula = Datos_Capacidad(y, 1)

'Determinamos la posición del aula en la matriz de disponibilidad

For x = 1 To Au

If Aula = Disponibilidad(x, 1) Then

If turnoal = 1 Then

Posaula = x

x = Au

ElseIf turnoal = 2 Then

Posaula = x + Au

x = Au

End If

End If

Next x

'Comprobamos que el aula esté disponible en el día de destino

If Disponibilidad(Posaula, diadest + 1) = 0 Then

.....

''' ASIGNATURA CABE EN UN AULA '''

.....

'Si el aula está disponible y tiene suficiente capacidad para albergar a los alumnos matriculados, se asigna ese aula

If Datos_Capacidad(y, 3) >= matal Then

If turnoal = 1 Then

auladestino(Posaula) = 1

ElseIf turnoal = 2 Then

auladestino(Posaula - Au) = 1

End If

y = Au

.....

''' ASIGNATURA NO CABE EN UN AULA '''

.....

ElseIf Datos_Capacidad(y, 3) < matal Then

'Si no caben en un aula, se asigna la primer aula analizada igualmente:

If turnoal = 1 Then

```

        auladestino(Posaula) = 1
    ElseIf turnoal = 2 Then
        auladestino(Posaula - Au) = 1
    End If
    Disponibilidad(Posaula, diadest + 1) = 1
    matalaux = matal - Datos_Capacidad(y, 3) 'He creado
matalaux en la que se irán descontando los alumnos para no perder el valor de
matal.

    'Y ahora se van a asignar las aulas más cercanas
para completar el número de matriculados.
    Do While matalaux > 0
        min = 10000
        For x = 1 To AuPar
            If Datos_Distancias(x, 1) = Aula Then
                Aula2 = Datos_Distancias(x, 2)
                'Localizamos en que fila de la tabla
disponibilidad está el aula2

                For k = 1 To Au
                    If Aula2 = Disponibilidad(k, 1)
Then
                        If turnoal = 1 Then
                            PosaulaD = k
                            k = Au
                        ElseIf turnoal = 2 Then
                            PosaulaD = k + Au
                            k = Au
                        End If
                    End If
                Next k
                'Localizamos en que fila de la tabla
capacidades está el aula2

                For k = 1 To Au
                    If Aula2 = Datos_Capacidad(k, 1)
Then
                        PosaulaC = k
                        k = Au
                    End If
                Next k

                'En el caso que tenga menor distancia,
y este disponible, se guarda el nuevo valor mínimo de distancia y el aula (en
mejor)

                    If Disponibilidad(PosaulaD, diadest +
1) = 0 And Datos_Distancias(x, 3) < min Then

```

```

min = Datos_Distancias(x, 3)
PosaulaDmejor = PosaulaD
PosaulaCmejor = PosaulaC
aulamejor = Aula2

ElseIf Disponibilidad(PosaulaD, diadest
+ 1) = 0 And Datos_Distancias(x, 3) = min Then

'En el caso de empate por distancia
entre Aula2 (nueva aula candidata) y aulamejor (mejor aula candidata hasta el
momento)

'Se escoge la que mejor se ajuste a
los alumnos que quedan por ubicar

If Abs(Datos_Capacidad(PosaulaC, 3)
- matalaux) < Abs(Datos_Capacidad(PosaulaCmejor, 3) - matalaux) Then

min = Datos_Distancias(x, 3)
PosaulaDmejor = PosaulaD
PosaulaCmejor = PosaulaC
aulamejor = Aula2

End If

End If

End If

Next x

'Como resultado de este for tendré el primer
aula seleccionada para completar la solución.

'Ahora guardo el aula seleccionada en el vector
donde registro las aulas utilizadas en esta asignatura

Disponibilidad(PosaulaDmejor, diadest + 1) = 1

If turnoal = 1 Then

auladestino(PosaulaDmejor) = 1

ElseIf turnoal = 2 Then

auladestino(PosaulaDmejor - Au) = 1

End If

'y actualizo el número de alumnos que quedan por
ubicar en este aula

matalaux = matalaux -
Datos_Capacidad(PosaulaCmejor, 3)

Loop

y = Au

End If

End If

Next y

```

.....

```

''' COMPROBAR SI MEJORA LA SOLUCIÓN '''
.....

'Como resultado en este punto tendríamos el vector aulaDestino
en el que tenemos registrado con 1 las nuevas aulas que se asignarían en el destino
'y en aulaOrigen tendríamos las aulas asignadas en el día inicial

.....

''' EVALUAR SOLUCIÓN ORIGEN '''
.....

distin = 0
For y = 1 To Au - 1
    If aulaOrigen(y) = 1 Then
        Aula1 = Disponibilidad(y, 1) 'veo que aula es la 1ª
        For i = y + 1 To Au
            If aulaOrigen(i) = 1 Then
                Aula2 = Disponibilidad(i, 1) 'veo que aula es
la 2ª
                For x = 1 To AuPar
                    If Datos_Distancias(x, 1) = Aula1 And
Datos_Distancias(x, 2) = Aula2 Then
                        distin = distin + Datos_Distancias(x,
3)
                    End If
                Next x
            End If
        Next i
    End If
Next y

.....

''' EVALUAR SOLUCIÓN DESTINO '''
.....

distfin = 0
For y = 1 To Au - 1
    If aulaDestino(y) = 1 Then
        Aula1 = Disponibilidad(y, 1) 'veo que aula es la 1ª
        For i = y + 1 To Au
            If aulaDestino(y) = 1 And aulaDestino(i) = 1 Then

```



```

Aula2 = Disponibilidad(i, 1) 'veo que aula es
la 2°
For x = 1 To AuPar
    If Datos_Distancias(x, 1) = Aula1 And
Datos_Distancias(x, 2) = Aula2 Then
        distfin = distfin + Datos_Distancias(x,
3)
        x = AuPar
    End If
Next x
End If
Next i
End If
Next y

'Comparación para ver si hacer el cambio
If distfin < distin Then

.....
''' BORRAR SOLUCIÓN DÍA ORIGEN '''
.....

'Y en el caso de mejorar por último faltaría actualizar la
solución

'Primero lo borro del día de origen
For y = 1 To Au
    If turnoal = 1 Then
        k = y
    ElseIf turnoal = 2 Then
        k = y + Au
    End If
    If aulaorigen(y) = 1 Then
        Disponibilidad(k, diaal + 1) = 0 'El aula origen
vuelve a estar disponible
        Horario_Solucion(k, diaal) = "" 'En horario solución
se vacía la casilla
        Aula = Disponibilidad(k, 1)
        'La capacidad total del día y turno se incrementa
en la capacidad del aula que hemos liberado
        For x = 1 To Au
            If Datos_Capacidad(x, 1) = Aula Then

```

```

                                capacidadtotal(turnoal,        diaal)        =
capacidadtotal(turnoal, diaal) + Datos_Capacidad(x, 3)
                                x = Au
                                End If
                                Next x
                                End If
                                Next y
                                'Ese día ya estaría disponible para hacer cualquier examen
                                de ese título y curso. es decir, actualizar compatibilidad (solo si no es
                                optativa)

                                If opt = 0 Then
                                    If NEq = 1 Then 'Si no tiene equivalentes, es decir,
                                    solo es una asignatura.

                                        For y = 1 To Asi
                                            If nombrealL = Datos_Asignaturas(y, 4) Then
                                                For x = 1 To Aux1
                                                    If      Compatibilidad(x,        1)        =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diaal And Compatibilidad(x, 5) = Datos_Asignaturas(y, 6)
Then
                                                        Compatibilidad(x, 4) = 0 'Se pone
que ya es compatible, porque hemos quitado la asignatura de este día.
                                                        x = Aux1
                                                        End If
                                                    Next x
                                                End If
                                            Next y
                                        End If
                                    Else 'Si que tiene asignaturas equivalentes.
                                        For j = 0 To NEq - 1
                                            For y = 1 To Asi
                                                If asigalL(j) = Datos_Asignaturas(y, 4)
Then
                                                        For x = 1 To Aux1
                                                            If      Compatibilidad(x,        1)        =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diaal And Compatibilidad(x, 5) = Datos_Asignaturas(y, 6)
Then
                                                                    Compatibilidad(x, 4) = 0
                                                                    x = Aux1
                                                                    End If
                                                            Next x
                                                            y = Asi
                                                        End If

```



```

        If nombrealL = Datos_Asignaturas(y, 4) Then
            For x = 1 To Aux1
                If Compatibilidad(x, 1) =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diadest And Compatibilidad(x, 5) = Datos_Asignaturas(y,
6) Then

                    Compatibilidad(x, 4) = 1 'Se pone
que ya NO es compatible

                    x = Aux1
                End If
            Next x
        End If
    Next y
Else 'Si que tiene asignaturas equivalentes.
    For j = 0 To NEq - 1
        For y = 1 To Asi
            If asigall(j) = Datos_Asignaturas(y, 4)
Then
                For x = 1 To Aux1
                    If Compatibilidad(x, 1) =
Datos_Asignaturas(y, 1) And Compatibilidad(x, 2) = Datos_Asignaturas(y, 2) And
Compatibilidad(x, 3) = diadest And Compatibilidad(x, 5) = Datos_Asignaturas(y,
6) Then

                        Compatibilidad(x, 4) = 1
                        x = Aux1
                    End If
                Next x
                y = Asi
            End If
        Next y
    Next j
End If
End If
End If
'He tenido que poner cuando seleccionaba las aulas de destino,
que su disponibilidad pasaba a ser 0.
'Sino escogia x veces la misma aula hasta completar los
matriculados.
'Entonces ahora, si no realizamos el cambio de aula porque
distfin > distin, habrá que volver a poner las aulas destino disponibles
If distfin > distin Then
    For y = 1 To Au
        If turnoal = 1 Then

```

```

        k = y
    ElseIf turnoal = 2 Then
        k = y + Au
    End If
    If auladestino(y) = 1 Then
        Disponibilidad(k, diadest + 1) = 0 'El aula destino
sí que está disponible
    End If
Next y
End If
End If
End If
Next iteración

```

```

.....
''' IMPRIMIR SOLUCIÓN '''
.....
For x = 1 To 2 * Au
    For y = 1 To numdias
        Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(x + 2, y + 2).Value =
Horario_Solucion(x, y)
    Next y
Next x

```

```

.....
''' EVALUACIÓN SOLUCIÓN '''
.....
distanciatotal = 0
If turno = 0 Then
    nfil = 2 * Au
Else
    nfil = Au
End If
For t = 1 To numdias
    For i = 1 To nfil - 1
        For j = i + 1 To nfil
            If Horario_Solucion(i, t) <> "" Then

```

If Horario_Solucion(i, t) = Horario_Solucion(j, t) Then 'Si coincide la asignatura programada en dos aulas hay que tener en cuenta su distancia.

```

        Aula1 = Disponibilidad(i, 1)
        Aula2 = Disponibilidad(j, 1)
        For x = 1 To AuPar
            If Aula1 = Datos_Distancias(x, 1) And Aula2 =
Datos_Distancias(x, 2) Then
                distanciatotal = distanciatotal +
Datos_Distancias(x, 3)
            End If
        Next x
    End If
End If
Next j
Next i
Next t
Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(3, 14).Value = distanciatotal

```

```

holguratotalsolucion = 0
capacidadtotalsolucion = 0
matriculadostotalsolucion = 0
If turno = 0 Then
    numturnos = 2
Else
    numturnos = 1
End If
For x = 1 To Au
    For y = 1 To Au * numturnos
        If Datos_Capacidad(x, 1) = Disponibilidad(y, 1) Then
            For t = 1 To numdias
                capacidadtotalsolucion = capacidadtotalsolucion +
Datos_Capacidad(x, 3) * Disponibilidad(y, t + 1)
            Next t
        End If
    Next y
Next x
For x = 1 To Asi
    If cuatrimestre = Datos_Asignaturas(x, 3) Then

```

```
        matriculadostotalsolucion      =      matriculadostotalsolucion      +
Datos_Asignaturas(x, 5)
    End If
Next x
holguratotalsolucion = capacidadtotalsolucion - matriculadostotalsolucion
Worksheets("HORARIO SOLUCIÓN CUATRI").Cells(6, 14).Value = holguratotalsolucion

End Sub
```

Anexo VI. Código de la función GenerarAleatorio

```
Function GenerarAleatorio(min As Integer, max As Integer) As Integer

    'Randomize (i)

    Dim numero As Integer

    numero = Int((max - min + 1) * Rnd() + min)

    GenerarAleatorio = numero

End Function
```


Anexo VII. Horario Solución

En este apartado del anexo se presentan las soluciones de la heurística constructiva y de la heurística de búsqueda local para 5 iter., 10 iter., 50 iter., 100 iter., 250 iter., 500 iter., 750 iter., 1000 iter. y 5000 iter. Todas ellas han sido aplicadas a un periodo de exámenes del cuatrimestre A, con ambos turnos y con un periodo de exámenes formado por 10 días.

DÍA	AULA	1	2	3	4	5	6	7	8	9	10	
TURNO MAÑANA	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 023	33809	33804	11402	33805	33813	11404	34079	11473	11490	10169	
	AULA 024				34368			34369				
	AULA 025											12950
	AULA 031	11470	12929	11475	12932	11407	11426	11529	11416	11408	11418	11423
	AULA 032	11470	12929	11475	12932	11407	11426	11529	11416	11408	11418	11423
	AULA 033	11499	12262	12934	11472	11407	11426	11529	11416	11408	11418	11423
	AULA 034					11407	11426	11529	11416	11408	11418	11423
	AULA 035					11407	11426	11529	11416	11408	11418	11423
	AULA 041	11499	12262	12934	11472	11407	11426	11529	11416	11473	11490	10169
	AULA 110						34358					
	AULA 111	3386	33786	3474	33904	34378	33902		33910	14002	34376	34081
	AULA 112											
	AULA 211											
	AULA 212				14661		14666			13036		34382
	AULA 213											
	AULA 214	13444	30369	32603	30455			32604				
	AULA 215	33905	34373	32802	35125	33900	34371		34377	33903	33901	33898
	AULA 222					11478						
	AULA 223											
	AULA 311	34935	34934	33906				14240	34555	34931		
	AULA 312	12265	13034	11494	12268	13040	13040	13963	11497	34381	14004	12297
AULA 323	12265	13034	11494	12268	13040	13040	13963	11497	13036	12285	12297	
AULA 410	13755	13884	34083	34082	34938	34938			34084	34932	34933	
AULA 411	14011	32807	34372	14662	14665	14009	31820	30459	31817	31824	14025	
AULA 412	34385					14663	35372	35127		35128		
AULA 420	32804	35392	35394	32803	32803	31851	35391	32801	35393	30384	32800	
AULA 421	13045	13058	13917	11498	12287	12287	12956	13050	13054	13053	13066	
AULA 424	13045	13058	13917	11498	12287	12287	12956	13050	13054	13053	13066	
AULA 425											12956	
AULA 521	12275	12274	12291	12941	12946	12946	33465	12951	12937	33470	33465	
AULA 522	12275	12274	12291	12941	12946	12946	33465	12951	12937	33470	33465	
AULA 523	34367	33473	33464	33468								
AULA 524				14020		30381	14030	14022			32606	
AULA 525	34367	33473	33464	33468								
AULA 526												
AULA 527												

Anexo VII. Figura 1. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)

TURNO TARDE													
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 022	33749												
AULA 023													
AULA 024							33706		33476				
AULA 025													
AULA 031	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 032	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 033	11421;11524	11506	11505	12954	12949	12266	12963	11489	12942	12960			
AULA 034	11421;11524								33707				
AULA 035	11421;11524												
AULA 041	33749	11506	11505	12954	12949	12266	12963	11489	12942	12960			
AULA 110													
AULA 111	33764	33717	33683	33682	31823	33732;33741	33763	33478	33485	33695			
AULA 112													
AULA 211													
AULA 212	33684												
AULA 213	33480												
AULA 214													
AULA 215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481			
AULA 222													
AULA 223													
AULA 311													
AULA 312	12961	12272	33479	13032	33718	33486	33488	33752	33719	33714			
AULA 323	12961	12272	12273	13032	33681	33486	33757	33723	33487	33733			
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688			
AULA 412	33727			33724	33756	33728	33737	33739	33740				
AULA 420	33729	33491	12273	33730	33483	13059	12280	33489	34762	31826			
AULA 421	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 424	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 425													
AULA 521		13062	13033	13068	12288	12298	13051	12957		12294			
AULA 522		13062	13033	13068	12288	12298	13051	12957		12294			
AULA 523		13041	12947	33705;35387		12939							
AULA 524	11487	33678	33722	33753	33694	33755	33721	33720	33761	33760			
AULA 525		13041	12947	33705;35387		12939							
AULA 526													
AULA 527	33738								33679				

Anexo VII. Figura 2. Horario Solución Heurística Constructiva para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33811	33803	33814	11398	11405	33807	11399	11403	33806	33808
	AULA 012	33811	33803	33814	11398	11405	33807	11399	11403	33806	33808
	AULA 013	33811	33803	33814	11398	11405	33807	11399	11403	33806	33808
	AULA 014	33811	33803	33814	11398	11405	33807	11399	11403	33806	33808
	AULA 021	33811	33803	33814	11398	11405	33807	11399	11403	33806	33808
	AULA 022	33811	33803	33814	11398	11405	33807	11399	11403	33806	33808
	AULA 023	33811	33803		11468	12930	12927	11474		12260	
	AULA 024			30456	13441	30457			32602	13438	
	AULA 025										
	AULA 031	11469	12931	11401	11431;11533	11429;11536	11433	11420;11531	11422;11532	11430;11530	12933
	AULA 032	11469	12931	11401	11431;11533	11429;11536	11433	11420;11531	11422;11532	11430;11530	12933
	AULA 033	11469	11471	11401	11431;11533	11429;11536	11433	11420;11531	11422;11532	11430;11530	12933
	AULA 034	12928	12945	11401	11431;11533	11429;11536	11433	11420;11531	11422;11532	11430;11530	11476
	AULA 035			11401	11431;11533	11429;11536	11433	11420;11531	11422;11532	11430;11530	12953
	AULA 041	12928	11471	11401	11468	12930	12927	11474	11422;11532	12260	11476
	AULA 110								12964		
	AULA 111	13233;14269	12311	33896	12309	34554	33907	33908	34375	13238	34553
	AULA 112	12980	33776	12970	13234	11500	34383	33499	13752;13757	35388;33709	35133
	AULA 211	12972	14686	12975	12981		33792	15028	12982	12983	
	AULA 212	33780	11490	12971	31818	14005	14000	31830	34253;34256	14003	12261
	AULA 213	13998	32806	14014	12969	12318	13027	14023	14007	14685	12310
	AULA 214	13751;13756	33778	12974	12325	33772	32605	13237	12978	33769;34764	14060;13757
	AULA 215	13440	12323	33775	34380	32809	13442	32607	12979	34379	34374
	AULA 222										
	AULA 223										
	AULA 311	34937	14019	33800;33504	34251;34255	13232;14270	35389;33708	3235;14664;1466	13445	13239	13028
	AULA 312	13918	11490	11493	13038	34254	13753	13754	12267	12952	12261
	AULA 323	13918	11490	11493	13038	13985	12271	13042	12267	12952	
	AULA 410	33909	33897	33798;33712	33911	33899	13242	14272	34936	33794;34765	33912
	AULA 411	33783	13020	32805	12973	13021	33781	13022	31829	32609	31828
	AULA 412	33779	33791	33789	12326	33777	34252	34384	34767;33788	33784	33770
	AULA 420	32808	13024	13023	13038	13985	12271	13042	35390	31822	13025
	AULA 421	11496	12269	13037	13061	13985	12271	13042	13035	13067	12289
	AULA 424	11496	12269	13037	13061	13985	12271	13042	13035	13067	12289
	AULA 425										
	AULA 521	13064	13060	12281	12955	12279	11501	13069	12283	12292	13044
	AULA 522	13064	13060	12281	12955	12279	11501	13069	12283	12292	13044
	AULA 523	33475	33472			33469	33467	33474			33471
	AULA 524	30463	13236;12977	12313	13437	14413;14414	33782	14412	33790	13240	32608
	AULA 525	33475	33472			33469	33467	33474			33471
	AULA 526										
	AULA 527	33501	11495	33771;34763	12316	33785	33793	14415	13026	33795	12976

TURNO MAÑANA

Anexo VII. Figura 3. Horario Solución Heurística Constructiva para el Cuatrimestre B (turno de mañana) (Fuente: Elaboración propia)

AULA 011	33716	11413	11434	11414	11412	11432,11535	11410	11409	11491	11479
AULA 012	33716	11413	11434	11414	11412	11432,11535	11410	11409	11491	11479
AULA 013	33716	11413	11434	11414	11412	11432,11535	11410	11409		
AULA 014	33716	11413	11434	11414	11412	11432,11535	11410	11409		
AULA 021	33716	11413	11434	11414	11412	11432,11535	11410	11409	12284	12943
AULA 022	33716	11413				11432,11535			12284	12943
AULA 023	33716					11432,11535				
AULA 024	33716	11511	34036	34092	34093		34100	34097	34103	34089
AULA 025										
AULA 031	12278	12936	11480	11484	11482	13057	11483	12944	33745	11521
AULA 032		12936	11480	11484	11482	13057	11483	12944	33745	11521
AULA 033			12277	12968	33751				12938	12286
AULA 034	34087	11515						11509		
AULA 036										
AULA 041			12277	12968	33751		13030		12938	12286
AULA 110										
AULA 111	34104	36396,33758	33680	33742	33767,33712	33895	33715	33711,33766,33797	14006	3710,33765,33797
AULA 112										
AULA 211										
AULA 212										
AULA 213										
AULA 214			11510							
AULA 215	33696	33699	14001	14008	33692	33495	34085		33494	34086
AULA 222										
AULA 223										
AULA 311				34547				11516		
AULA 312										
AULA 323	13065	13052				34094	34102		34098	33725
AULA 410					34088	13989	12940	12290	13063	11517
AULA 411	14024	33736	33693	33493	33726	14017	14021	33685	33894	14027
AULA 412	33497	33496	11523			33754	33743	34090		34096
AULA 420	35396	14015	33734	14028	14010	34099	34091	14026	14029	33492
AULA 421										
AULA 424										
AULA 425										
AULA 521										
AULA 522										
AULA 523										
AULA 524										
AULA 525										
AULA 526										
AULA 527										

TURNO TARDE

Anexo VII. Figura 4. Horario Solución Heurística Constructiva para el Cuatrimestre B (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10	
TURNO MAÑANA	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 023	33809	33804	11402	33805	33813	11404	34079	11473	11490	10169	
	AULA 024					34368		34369	34370			
	AULA 025										12950	
	AULA 031	11470	12929	11475	12932	11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499	12262	12934	11472	11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034					11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035					11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499	12262	12934	11472	11407	11407	11426;11529	11416	11473	11490	10169
	AULA 110					34358						
	AULA 111	35386;33786;34766	14013	33904	33904	34378	33902	33902	33910	14002	34376	34081
	AULA 112											
	AULA 211											
	AULA 212				14661	14666				13036		34382
	AULA 213											
	AULA 214	13444	30369	32603	30455			32604				
	AULA 215	33906	34373	32802	35126	33900	33900	34371	34377	33903	33901	33898
	AULA 222					11478						
	AULA 223											
	AULA 311	34935	34934	33306				14240	34555	34931		
	AULA 312	12266	13034	11494	12268	13040	13040	13963	11497	34381	14004	12297
	AULA 323	12266	13034	11494	12268	13040	13040	13963	11497	13036	12285	12297
	AULA 410	13755	13884	34083	34082	34938				34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663	14663	35372	35127		35128	
	AULA 420	32804	35392	35394	32803	31831	31831	35391	32801	35393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12287	12956	13050	13054	13053	13066
	AULA 424	13045	13058	13917	11498	12287	12287	12956	13050	13054	13053	13066
	AULA 425										12956	
	AULA 521	12275	12274	12291	12941	12946	12946	33465	12951	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	12946	33465	12951	12937	33470	33466
	AULA 523	34367	33473	33464	33468							
	AULA 524				14020	30381		14030	14022			32606
	AULA 525	34367	33473	33464	33468							
	AULA 526											
	AULA 527											

Anexo VII. Figura 5. Horario Solución Algoritmo de Búsqueda Total con 5 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)

TURNO TARDE													
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 022	33749												
AULA 023													
AULA 024							33706		33476				
AULA 025													
AULA 031	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 032	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 033	11421;11524	11506	11505	12564		12266	12963	11489	12942	12960			
AULA 034	11421;11524								33707				
AULA 035	11421;11524												
AULA 041	33749	11506	11505	12564		12266	12963	11489	12942	12960			
AULA 110													
AULA 111	33764	33717	33688	33682	31823	33732;33741	33763	33478	33485	33695			
AULA 112													
AULA 211													
AULA 212	33684				33718		33488	33752	33719	33714			
AULA 213	33480				33681		33757	33723	33487	33733			
AULA 214													
AULA 215	33689	34761	33701	33762	33686	31825	33691	33630	33687	33481			
AULA 222													
AULA 223													
AULA 311					33703			33702	33736	33704			
AULA 312	12961	12272	33479	13032	12295	33486	31827	13031	13029	33744			
AULA 323	12961	12272	12273	13032	12295	13059	12280	13031	13029	33744			
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33484	33430	33759	33657	33688			
AULA 412	33727			33724	33756	33728	33737	33739		33740			
AULA 420	33729	33491	12273	33730	33483	13059	12280	33489	34762	31826			
AULA 421	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 424	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 425													
AULA 521		13062	13033	13068	12288	12298	13051	12957		12294			
AULA 522		13062	13033	13068	12288	12298	13051	12957		12294			
AULA 523		13041	12947	33705;33738		12939	12949						
AULA 524	11487	33678	33722	33753	33694	33755	33721	33720	33761	33760			
AULA 525		13041	12947	33705;33738		12939	12949						
AULA 526													
AULA 527	33738								33679				

Anexo VII. Figura 6. Horario Solución Algoritmo de Búsqueda Total con 5 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 023	33809	33804	11402	33805	33813	11404	34079	11473	11490	10169
	AULA 024				34368			34369	34370		
	AULA 025										12950
	AULA 031	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499	12262	12934	11472	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499	12262	12934	11472	11407	11426;11529	11416	11473	11490	10169
	AULA 110						34358				
	AULA 111	35386;33786;34766	14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112										
	AULA 211										
	AULA 212				14661	14666			13036	12285	34382
	AULA 213										
	AULA 214	13444	30369	32603	30455		32604				
	AULA 215	33905	34373	32802	35126	33900	34371	34377	33903	33901	33898
	AULA 222					11478					
	AULA 223										
	AULA 311	34935	34934	33906			14240	34555	34931		
	AULA 312	12265	13034	11494	12268	13040	13963	11497	34381	14004	12297
	AULA 323	12265	13034	11494	12268	13040	13963	11497	13036	12285	12297
	AULA 410	13755	13884	34083	34082	34938			34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663		35127		35128	
	AULA 420	32804	35392	35394	32803	31831	35372	32801	35393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12956	13050	13054	13053	13066
	AULA 424	13045	13058	13917	11498	12287	12956	13050	13054	13053	13066
	AULA 425									12936	
	AULA 521	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 523	34367	33473	33464	33468						
	AULA 524				14020	30381	14030	14022			32606
	AULA 525	34367	33473	33464	33468						
	AULA 526										
	AULA 527										

TURNO MAÑANA

Anexo VII. Figura 7. Horario Solución Algoritmo de Búsqueda Total con 10 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)

TURNO TARDE													
AULA 011	11435	11417	11411	11428,11526	11427,11534	11436	11419	11424,11525	11425,11527	11415			
AULA 012	11435	11417	11411	11428,11526	11427,11534	11436	11419	11424,11525	11425,11527	11415			
AULA 013	11435	11417	11411	11428,11526	11427,11534	11436	11419	11424,11525	11425,11527	11415			
AULA 014	11435	11417	11411	11428,11526	11427,11534	11436	11419	11424,11525	11425,11527	11415			
AULA 021	11435	11417	11411	11428,11526	11427,11534	11436	11419	11424,11525	11425,11527	11415			
AULA 022	33749												
AULA 023													
AULA 024							33706		33476				
AULA 025													
AULA 031	11421,11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 032	11421,11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 033	11421,11524	11506		12954	12949	12266	12963	11489	12942	12960			
AULA 034	11421,11524								33707				
AULA 035	11421,11524												
AULA 041	33749	11506		12954	12949	12266	12963	11489	12942	12960			
AULA 110													
AULA 111	33764	33717	33683	33682	31823	33732,33741	33763	33478	33485	33695			
AULA 112													
AULA 211													
AULA 212	33684				33718		33488	33752	33719	33714			
AULA 213	33480				33681		33757	33723	33487	33733			
AULA 214													
AULA 215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481			
AULA 222													
AULA 223													
AULA 311													
AULA 312	12961	12272	33479	13032	12295	33486	31827	33702	33736	33704			
AULA 323	12961	12272		13032	12295	13059	12280	13031	13029	33744			
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688			
AULA 412	33727			33724	33756	33728	33737	33739		33740			
AULA 420	33729	33491		33730	33483	13059	12280	33489	34762	31826			
AULA 421	12282	33750	12276	13988	33747	33748	13056	12959	12948				
AULA 424	12282	33750	12276	13988	33747	33748	13056	12959	12948				
AULA 425				11505									
AULA 521		13062	13033	13068	12288	12298	13051	12957	33746	12294			
AULA 522		13062	13033	13068	12288	12298	13051	12957	33746	12294			
AULA 523		13041	12947	33705,33387		12959			12273				
AULA 524	11487	33678	33722	33753	33694	33755	33721	33720	33761	33760			
AULA 525		13041	12947	33705,33387		12959			12273				
AULA 526				11505									
AULA 527	33738								33679				

Anexo VII. Figura 8. Horario Solución Algoritmo de Búsqueda Total con 10 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33609	33604	11402	33605	33613	11404	33610	11406	11400	33612
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 023	33809	33804	11402	33805	33813	11404	34079	11473	11490	10169
	AULA 024					34368		34369	34370		
	AULA 025										12950
	AULA 031	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499	12262		11472	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499	12262		11472	11407	11426;11529	11416	11473	11490	10169
	AULA 110						34358				
	AULA 111	35386;33786;34766	14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112										
	AULA 211										
	AULA 212				14661	14666			13036	12285	34382
	AULA 213										
	AULA 214	13444	30369	32603	30455		32604				
	AULA 215	33905	34373	32802	35126	33900	34371	34377	33903	33901	33898
	AULA 222					11478					
	AULA 223										
	AULA 311	34935	34934	33906			14240	34555	34931		
	AULA 312	12265	13034		12268	13040	13963	11497	34381	14004	12297
	AULA 323	12265	13034		12268	13040	13963	11497	13036	12285	12297
	AULA 410	13755	13884	34033	34032	34938			34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663	35372	35127		35128	
	AULA 420	32804	35392	35394	32803	31831	35391	32801	35393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12956	13050	13054	13053	13066
	AULA 424	13045	13058	13917	11498	12287	12956	13050	13054	13053	13066
	AULA 425	12934								12936	
	AULA 521	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 523	34367	33473	33464	33468				11494		
	AULA 524				14020	30381	14030	14022			32606
	AULA 525	34367	33473	33464	33468				11494		
	AULA 526	12934									
	AULA 527										

Anexo VII. Figura 9. Horario Solución Algoritmo de Búsqueda Total con 50 iter. para el Cuatrimestre A (turno de mañana) (Fuente: Elaboración propia)

TURNO TARDE													
AUIA011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AUIA012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AUIA013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AUIA014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AUIA021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AUIA022	33749												
AUIA023							33706		33476				
AUIA024													
AUIA025													
AUIA031	11421;11524		11477	11485	11481	11486	11507	11488		11508			
AUIA032	11421;11524		11477	11485	11481	11486	11507	11488		11508			
AUIA033	11421;11524	11506	11506	12954	12949	12266		11489	12942	12960			
AUIA034	11421;11524								33707				
AUIA035	11421;11524												
AUIA041	33749	11506	11506	12954	12949	12266		11489	12942	12960			
AUIA110													
AUIA111	33764	33717	33683	33682	31823	33732;33741	33763	33478	33485	33635			
AUIA112													
AUIA211													
AUIA212	33684				33718		33488	33752	33719	33714			
AUIA213	33480				33681		33757	33723	33487	33733			
AUIA214													
AUIA215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481			
AUIA222													
AUIA223													
AUIA311													
AUIA312	12961	12272	33479	13032	12295	33486	31827	33702	33736	33704			
AUIA323	12961	12272	12273	13032	12295	13059	12280	13031	13029	33744			
AUIA410													
AUIA411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688			
AUIA412	33727			33724	33756	33728	33737	33759	33740	33740			
AUIA420	33729	33491	12273	33730	33483	13059	12280	33489	34762	31826			
AUIA421	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AUIA424	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AUIA425						12963							
AUIA521		13062	13033	13068	12288	12298	13051	12667	11492	12294			
AUIA522		13062	13033	13068	12288	12298	13051	12667	11492	12294			
AUIA523		13041	12947	33705;35387		12939		12293					
AUIA524	11487	33678	33722	33753	33694	33755	33721	33720	33761	33760			
AUIA525		13041	12947	33705;35387		12939		12293					
AUIA526						12963							
AUIA527	33738								33679				

Anexo VII. Figura 10. Horario Solución Algoritmo de Búsqueda Total con 50 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10	
TURNO MAÑANA	AULA 001	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 002	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 003	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 004	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 005	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 006	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 007	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 008	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 009	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 010	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 015	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 016	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 017	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 018	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 019	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 020	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 023	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 024	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 025	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 026	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 027	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 028	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 029	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 030	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 031	11470	12929	11475	12932	11407	11416	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11416	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499	12994	12994	11472	11407	11416	11426;11529	11416	11408	11418	11423;11528
	AULA 034	11499	12994	12994	11472	11407	11416	11426;11529	11416	11408	11418	11423;11528
	AULA 035	11499	12994	12994	11472	11407	11416	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499	12994	12994	11472	11407	11416	11426;11529	11416	11408	11418	11423;11528
	AULA 110	35386;33786;34766	14013	14016	33904	34378	33910	34376	34081			
	AULA 111											
	AULA 112											
	AULA 211											
	AULA 212				14661					13036	12285	34382
	AULA 213											
	AULA 214	13444	30369	32603	30455			32604				
	AULA 215	33905	34373	32802	35126	33900	34377	33903	33901			33898
	AULA 222					11478						
	AULA 223											
	AULA 311	34935	34934	33906				14240	34555	34931		
	AULA 312	12265	13034	11494	12268			13963	11497	34381	14004	12297
	AULA 323	12265	13034	11494	12268	33464		13963	11497	13036	12285	12297
	AULA 410	13755	13684	34083	34082	34938				34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009		31820	30459	31817	31824	14025
	AULA 412	34385	14666			14663		35372	35127		35128	
AULA 420	32804	35392	35394	32803	31831		35391	32801	35393	30384	32800	
AULA 421	13045	13058	13917	11498	12287		12956	13050	13054	13053	13066	
AULA 424	13045	13058	13917	11498	12287		12956	13050	13054	13053	13066	
AULA 425	13045	13058	13917	11498	12287		12956	13050	13054	13053	13066	
AULA 521	12275	12274	12291	12941	12946		33465	12951	12937	12936	33466	
AULA 522	12275	12274	12291	12941	12946		33465	12951	12937	12936	33466	
AULA 523	34367	33473		33468			13040	13040	33470	33470		
AULA 524				14020	30381		14030	14022			32606	
AULA 525	34367	33473		33468				13040				
AULA 526	11490	11490						12262				
AULA 527								12262				

Anexo VII. Figura 11. Horario Solución Algoritmo de Búsqueda Total con 100 iter. para el Cuatrimestre A (turno de mañana)
(Fuente: Elaboración propia)

TURNO TARDE													
AULA 011	13059	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 012	13059	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 013		11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 014		11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 021		11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 022	33749												
AULA 023							33706						
AULA 024													
AULA 025													
AULA 031	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 032	11421;11524	11492	11477	11485	11481	11486	11507	11488	12293	11508			
AULA 033	11421;11524	11506	11505		12949		12963	11489	12942				
AULA 034	11421;11524								33707				
AULA 035	11421;11524												
AULA 041	33749	11506	11505		12949		12963	11489	12942				
AULA 110													
AULA 111	33764	33717	33683	33682	31823	33732;33741	33763	33478	33485	33695			
AULA 112													
AULA 211													
AULA 212	33684				33718		33488	33752	33719	33714			
AULA 213	33480				33681		33757	33723	33487	33733			
AULA 214													
AULA 215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481			
AULA 222													
AULA 223													
AULA 311					33703			33702	33796	33704			
AULA 312	12961	12272	33479	13032	12295	33486	31827	13031	13029	33744			
AULA 323	12961	12272	12273	13032	12295			13031	13029	33744			
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688			
AULA 412	33727			33724	33756	33728	33737	33739		33740			
AULA 420	33729	33491	12273	33730	33483			33489	34762	31826			
AULA 421	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 424	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 425		12960			12954				11435				
AULA 521		13062	13033	13068	12288	12298	13051	12957	11435	12294			
AULA 522		13062	13033	13068	12288	12298	13051	12957	11435	12294			
AULA 523		13041	12947	33705;35387	12266	12939	12266	12280	11435				
AULA 524	11487	33678	33722	33753	33694	33755	33721	33720	33761	33760			
AULA 525		13041	12947	33705;35387		12939	12266	12280	11435				
AULA 526		12960			12954				11435				
AULA 527	33738								33679				

Anexo VII. Figura 12. Horario Solución Algoritmo de Búsqueda Total con 100 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 023	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812
	AULA 024					34368		34369	34370		
	AULA 025										12950
	AULA 031	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033		12262		11472	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035					11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041		12262		11472	11407	11426;11529	11416	33468	11490	
	AULA 110					13050	34358				
	AULA 111	35396;33786;34766	14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112					13050					
	AULA 211										
	AULA 212				14661	14666			13036		34362
	AULA 213										
	AULA 214	13444	30369	32603	30455		32604				
	AULA 215	33905	34373	32802	35126	33900	34371	34377	33903	33901	33898
	AULA 222					11478					
	AULA 223										
	AULA 311	34935	34934	33906			14240	34555			
	AULA 312	12265	13034	11494	30455	13040			34381	14004	12297
	AULA 323	12265	13034	11494	13040	13040			13036		12297
	AULA 410	13755	13884	34083	34082	34938		34931	34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663	35372	35127		35128	
	AULA 420	32804	35392	35394	32803	31831	35391	32801	35393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12956	12285	13054	13053	13056
	AULA 424	13045	13058	13917	11498	12287	12956	12285	13054	13053	13056
	AULA 425			10169		12268		13963		12936	
	AULA 521	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 523	34367	33473	33464	12934	12934	11497	11473	11473	11473	11499
	AULA 524				14020	30381	14030	14022		11473	32606
	AULA 525	34367	33473	33464	12934	12934	11497			11473	11499
	AULA 526			10169		12268		13963			
	AULA 527										11499

Anexo VII. Figura 13. Horario Solución Algoritmo de Búsqueda Total con 250 iter. para el Cuatrimestre A (turno de mañana)
(Fuente: Elaboración propia)

TURNOTARDE													
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 022													
AULA 023													
AULA 024							33706		33476				
AULA 025													
AULA 031	11421;11524	11492	11477	11485	11481	11486	11507	12942	12293	11508			
AULA 032	11421;11524	11492	11477	11485	11481	11486	11507	12942	12293	11508			
AULA 033	11421;11524	11506		12564	12949	12266		11489					
AULA 034	11421;11524								33707				
AULA 035	11421;11524												
AULA 041		11506		12564	12949	12266		11489					
AULA 110										13059			
AULA 111		33717	33683	33682	31823	33752;33741	33763	33478	33485	33656			
AULA 112										13059			
AULA 211													
AULA 212	33684				33718		33488	33752		33714			
AULA 213	33480				33681		33757	33723	33487	33733			
AULA 214													
AULA 215	33689	34761	33719	33762	33686	31825	33691	33690	33687	33481			
AULA 222		12295	11505							12963			
AULA 223		12295	11505							12963			
AULA 311					33703			33702	33736	33704			
AULA 312	12961		33479	13032		33486	31827	13031		33744			
AULA 323	12961		12273	13032				13031					
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688			
AULA 412	33727			33724	33756	33728	33737	33739		33740			
AULA 420	33729	33491	12273	33730	33483	33764	33701	33489	34762	31826			
AULA 421	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 424	12282	33750	12276	13988	33747	33748	13056	33746	12959	12948			
AULA 425		12960	33749			33748		12272		12280			
AULA 521		13062	13033	13068	12288	12298	13051	12567		12294			
AULA 522		13062	13033	13068	12288	12298	13051	12567		12294			
AULA 523		13041	12947	33705;35387	13029	12999	33721	33720	33761	11488			
AULA 524	11487	33678	33722	33753	33694	33755	33721	33720	33761	33760			
AULA 525		13041	12947	33705;35387	13029	12999				11488			
AULA 526		12960	33749					12272		12280			
AULA 527	33738								33679				

Anexo VII. Figura 14. Horario Solución Algoritmo de Búsqueda Total con 250 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 012	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 013	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 014	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 021	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 022	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 023	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 024	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 025	33809	33804	11402	33806	33813	11404	33810	11406	11400	33812
	AULA 031	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 032	11470	12929	11475	12932	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 033	11499		11407	11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 034	11499		11407	11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 035	11499		13066	11407	11407	11426;11529	11416	11408	11418	11423;11528
	AULA 041	11499			11407	11407	11426;11529	11416	11473		34367
	AULA 110	35386;33786;34766			34358	12268					
	AULA 111		14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112							12268			
	AULA 211										
	AULA 212				14661						34382
	AULA 213										
	AULA 214	13444	30369	32603	30455	33900	32604	34377	33903	33901	33898
	AULA 215	33906	34373	32802	35126	11478	12285	13040			
	AULA 222				12297		12285	13040			
	AULA 223				12297		12285	13040			
	AULA 311	34935	34934	33906		11472	14240	34555	34931		
	AULA 312			11494			13963		34381	14004	
	AULA 323	33473		11494		11472	13963				
	AULA 410	13755	13884	34083	34082	34938			34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	31820	30459	31817	31824	14025
	AULA 412	34385				14663	35372	35127	14666	35128	
	AULA 420	32804	35392	35394	32803	31831	35391	32801	35393	30384	32800
	AULA 421	13045	13058	13917	11498	12287	12956	13050	13054	13053	12262
	AULA 424	13045	13058	13917	11498	12287	12956	13050	13054	13053	12262
	AULA 425				13034		11497			12936	
	AULA 521	12275	12274	12291	12941	12946	33466	12561	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	33466	12561	12937	33470	33466
	AULA 523		11490	33464	33468	12934		10169	12265		13036
	AULA 524				14020	30381	14030	14022			32606
	AULA 525				33468			10169			
	AULA 526	11490	11490	33464	33468	12934		10169	12265		13036
	AULA 527				13034		11497				

TURNNO MAÑANA

Anexo VII. Figura 15. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de mañana)
(Fuente: Elaboración propia).

TURNO TARDE												
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415		
AULA 022												
AULA 023												
AULA 024							33706		33476			
AULA 025												
AULA 031	11421;11524	11492	11477	11485	13029	11486	11507	12273	12293			
AULA 032	11421;11524	11492	11477	11485	13029	11486	11507	12273	12293			
AULA 033	11421;11524	11506		12954				11489	12942			
AULA 034	11421;11524								33707			
AULA 035	11421;11524											
AULA 041		11506		12954		12282		11489	12942			
AULA 110												
AULA 111	33764	33717	33683	33682	31823	33752;33741	33763	33478		33656		
AULA 112												
AULA 211												
AULA 212	33684				33718		33488	33752	33719	33714		
AULA 213	33480				33681		33757	33723	33487	33733		
AULA 214												
AULA 215	33689	34761	33701		33686	31825	33691	33630	33687	33481		
AULA 222	13059											
AULA 223	13059											
AULA 311												
AULA 312	12961		12960		33703			33702	33736	33704		
AULA 313	12961		33479	13032	12256	33486	31827	12339	12280	33744		
AULA 323	12961		12960	13032	12256			12339	12280	33744		
AULA 410												
AULA 411	33482	33700	31821	33698	33477	33484	33490	33759	33697	33688		
AULA 412	33727			33724	33756	33728	33737	33739	33755	33740		
AULA 420	33729	33491	33762	33730	33483	33760	33485	33489	34762	31826		
AULA 421	12272	33750	12276	13988	33747	33748	13056	33746	12959	12948		
AULA 424	12272	33750	12276	13988	33747	33748	13056	33746	12959	12948		
AULA 425	12949	12963				13031			33749			
AULA 521	11508	13062	13033	13068	12288	12298	13051	12567	11481	12294		
AULA 522	11508	13062	13033	13068	12288	12298	13051	12567	11481	12294		
AULA 523		13041	12947	33705;35387	11488		12266		11505			
AULA 524	11487	33678	33722	33753	33694		33721	33720	33761			
AULA 525	12949	13041	12947	33705;35387	11488		12266		11505			
AULA 526	12949	12963				13031			33749			
AULA 527	33738								33679			

Anexo VII. Figura 16. Horario Solución Algoritmo de Búsqueda Total con 500 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10	
TURNO MAÑANA	AULA 011	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 012	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 013	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 014	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 021	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 022	33809	33804	11402	33805	33813	11404	33810	11406	11400	33812	
	AULA 023	33809	33804	11402	33805	33813	13040					
	AULA 024					34368	34079	34079	34369	34370		
	AULA 025						13040					12950
	AULA 031	11470	12929	11475	12952	11407	11407	11426;115 29	11416	11408	11418	11423;115 28
	AULA 032	11470	12929	11475	12952	11407	11407	11426;115 29	11416	11408	11418	11423;115 28
	AULA 033				11472	11407	11407	11426;115 29	11416	11408	11418	11423;115 28
	AULA 034					11407	11407	11426;115 29	11416	11408	11418	11423;115 28
	AULA 035	33468		33464		11407	11407	11426;115 29	11416	11408	11418	11423;115 28
	AULA 041	13050	13036		11472	11407	11407	11426;115 29	11416		34367	
	AULA 110	35386;33 786;34 766	14013	14016	33904	34378	33902	34358	12287			
	AULA 111								33910	14002	34376	34081
	AULA 112	13050	13036						12287			
	AULA 211											
	AULA 212				14661							34382
	AULA 213											
	AULA 214	13444	30369	32603	30455			32604				
	AULA 215	33905	34373	32802	35126	33900	33900	34371	34377	33903	33901	33898
	AULA 222	11494		10169		11478		12285			13034	
	AULA 223	11494		10169		12285		12285			13034	
	AULA 311	34935	12262	33906				14240	34555	34931	11473	
	AULA 312	12265			12268				11497	34381	14004	12297
	AULA 323	12265	12262		12268				11497		11473	12297
	AULA 410	13755	13884	34083	34082	34938	34938	34934	11497	34084	34932	34933
	AULA 411	14011	32807	34372	14662;14 665	14009	14009	31820	30459	31817	31824	14025
	AULA 412	34385			14666	14666	14663	35372	35127		35128	
	AULA 420	32804	35392	35394	33803	31831	31831	35391	32801	35393	30384	32800
	AULA 421	13045	1305 8	13917	11498			12566		13054	13053	13066
	AULA 424	13045	1305 8	13917	11498			12566		13054	13053	13066
	AULA 425		11499	13963					12934		12936	
	AULA 521	12275	12274	12291	12941			33465	12951	12937	33470	33466
	AULA 522	12275	12274	12291	12941			33465	12951	12937	33470	33466
	AULA 523		33473									11490
	AULA 524				14030			14030	14022			32606
	AULA 525		33473									11490
	AULA 526		11499	13963					12934			
	AULA 527		11499									

Anexo VII. Figura 17. Horario Solución Algoritmo de Búsqueda Total con 1000 iter. para el Cuatrimestre A (turno de mañana)
(Fuente: Elaboración propia)

TURNO TARDE													
AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415			
AULA 022													
AULA 023							33706		33476				
AULA 024													
AULA 025													
AULA 031	11421;11524	11492	11477	11485	11481	12295		12272	12293	11508			
AULA 032	11421;11524	11492	11477	11485	11481	12295		12272	12293	11508			
AULA 033	11421;11524				12949								
AULA 034	11421;11524				11507				33707				
AULA 035	11421;11524				11507								
AULA 041		12947	12939		12949								
AULA 110		11489			13056								
AULA 111	33764		33683	33682	31823	33732;33741	33763	33478	33485	33695			
AULA 112		11489			13056								
AULA 211													
AULA 212	33684						33488	33752	33719	33714			
AULA 213	33480						33757	33723	33487	33733			
AULA 214													
AULA 215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481			
AULA 222					12960								
AULA 223					12960								
AULA 311					33703	11506		33702	33736	33704			
AULA 312		12280	33479	13032		33486	31827	13031					
AULA 323		12280		13032		11506	12282	13031					
AULA 410													
AULA 411	33482	33700	31821	33698	33477	33678	33490	33759	33697	33688			
AULA 412	33727	33755	33738	33724	33756	33728	33737	33739		33740			
AULA 420	33729	33491	33718	33730	33483	33717	33484	33489	34762	31826			
AULA 421		33750	12276	13988	33747	33748	11486	33746	12959	12948			
AULA 424		33750	12276	13988	33747	33748	11486	33746	12959	12948			
AULA 425		12963	12954	33749	12266	33748	11486	33746	12273				
AULA 521	11505	13062	13033	13068	12288	12298	13051	12957		12294			
AULA 522	11505	13062	13033	13068	12288	12298	13051	12957		12294			
AULA 523		13029	33744	33705;35387	12288	12298	12961	12942	13059	11488			
AULA 524	11487		33722	33753	33694		33721	33720	33761	33760			
AULA 525		13029	33744	33705;35387			12961	12942	13059	11488			
AULA 526		12963	12954	33749	12266			12273					
AULA 527									33679				

Anexo VII. Figura 18. Horario Solución Algoritmo de Búsqueda Total con 1000 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)

DÍA	AULA	1	2	3	4	5	6	7	8	9	10
	AULA 011	33809	33804	11402	33806	33813	11404	33810	11406	11400	12297
	AULA 012	33809	33804	11402	33806	33813	11404	33810	11406	11400	12297
	AULA 013	33809	33804	11402	33806	33813	11404	33810	11406	11400	
	AULA 014	33809	33804	11402	33806	33813	11404	33810	11406	11400	
	AULA 021	33809	33804	11402	33806	33813	11404	33810	11406	11400	11494
	AULA 022	33809	33804	11402	33806	33813	11404	33810	11406	11400	11494
	AULA 023	33809	33804	11402	33806	33813	11473	34369	34370	34079	
	AULA 024	13040		11402	33806	34368	11473				
	AULA 025	13040					11473				12950
	AULA 031	11470	12929	11475	12932	11407	11426;115 29	11416	11408	11418	10169
	AULA 032	11470	12929	11475	12932	11407	11426;115 29	11416	11408	11418	10169
	AULA 033					11407	11426;115 29	11416	11408	11418	
	AULA 034		12956			11407	11426;115 29	11416	11408	11418	
	AULA 035		12956			11407	11426;115 29	11416	11408	11418	
	AULA 041	13066				11407	11426;115 29	11416			
	AULA 110	11497	13045				34368				
	AULA 111	35386;33786;34766	14013	14016	33904	34378	33902	33910	14002	34376	34081
	AULA 112	11497	13045				33812				
	AULA 211						33812				
	AULA 212				14661		33812				34382
	AULA 213						33812				
	AULA 214	13444	30369	32603	30455		32604				
	AULA 215	33906	34373	32802	36126	33900	34371	34377	33903	33901	33898
	AULA 222		11490			11478	12934				
	AULA 223		11490				12934				
	AULA 311		34934	33906			14240	34555		14004	
	AULA 312								34381		
	AULA 323	33473	34367	33468		33464					
	AULA 410	13765	13884	34083	34082	34938	34935	34931	34084	34932	34933
	AULA 411	14011	32807	34372	14662;14665	14009	31820	30459	31817	31824	14025
	AULA 412	34385		14666		14663	35372	35127		35128	
	AULA 420	32804	35392	35394	32803	31831	35391	32801	35393	30384	32800
	AULA 421			13917	11498	12287		13050	13054		
	AULA 424			13917	11498	12287		13050	13054		
	AULA 425		12265				11499			12936	
	AULA 521	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 522	12275	12274	12291	12941	12946	33465	12951	12937	33470	33466
	AULA 523					12262		13034		13963	11472
	AULA 524				14000	30381	14030	14022			32606
	AULA 525					12262		13034		13963	11472
	AULA 526		12265			12262	11499			13053	
	AULA 527					12262	11499			13053	11472

Anexo VII. Figura 19. Horario Solución Algoritmo de Búsqueda Total con 5000 iter. para el Cuatrimestre A (turno de mañana)
(Fuente: Elaboración propia)

AULA 011	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415
AULA 012	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415
AULA 013	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415
AULA 014	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415
AULA 021	11435	11417	11411	11428;11526	11427;11534	11436	11419	11424;11525	11425;11527	11415
AULA 022			33749	11489	33750		11505	12276		
AULA 023			33749	11489	33750		11505	12276	33476	
AULA 024			12282				33706			
AULA 025			12282							
AULA 031	11421;11524	11492	11477	11485	11481	11486	12961	11488		11508
AULA 032	11421;11524	11492	11477	11485	11481	11486	12961	11488		11508
AULA 033	11421;11524									
AULA 034	11421;11524	11506		13032	13031		13988	11507	33707	
AULA 035	11421;11524	11506		13032	13031		13988	11507		
AULA 041			33705;33887							
AULA 110		12280	12273	33747	12949					
AULA 111	33764	33717	33683	33682	31823	33732;33741	33763	33478	33485	33635
AULA 112		12280	12273	33747	12949					
AULA 211										
AULA 212	33684				33718		33488		33719	33714
AULA 213					33681		33757	33723	33487	33733
AULA 214										
AULA 215	33689	34761	33701	33762	33686	31825	33691	33690	33687	33481
AULA 222		12266		12942						
AULA 223		12266		12942						
AULA 311			12963		33703			33702	33736	33704
AULA 312			33479				31827		13029	
AULA 323	13041		12963				12947	12939	13029	
AULA 410										
AULA 411	33482	33700	31821	33698	33480	32484	33490	33759	33697	33688
AULA 412	33727	33738	33763	33724	33756	33728	33737	33739	33755	33740
AULA 420	33729	33491	33477	33730	33483	33752	33722	33489	34762	31826
AULA 421				12235	12293	33748	13566	33746	12959	12948
AULA 424				12235	12293	33748	13566	33746	12959	12948
AULA 425		12954					33744			
AULA 521		13062	13033	13068	12288	12298	13561	12957	12272	12294
AULA 522		13062	13033	13068	12288	12298	13561	12957	12272	12294
AULA 523					12960			13059		
AULA 524	11487	33678			33694		33721	33720	33761	33760
AULA 525					12960			13059		
AULA 526		12954					33744			
AULA 527									33679	

TURNO TARDE

Anexo VII. Figura 20. Horario Solución Algoritmo de Búsqueda Total con 5000 iter. para el Cuatrimestre A (turno de tarde) (Fuente: Elaboración propia)



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCOLA TÈCNICA
SUPERIOR ENGINYERIA
INDUSTRIAL VALÈNCIA

PRESUPUESTO

DESARROLLO E IMPLEMENTACIÓN DE HERRAMIENTAS HEURÍSTICAS PARA
LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES DE LA ETSII
CONSIDERANDO LA DISTANCIA ENTRE AULAS

TRABAJO FIN DE GRADO EN GRADO EN INGENIERÍA DE ORGANIZACIÓN
INDUSTRIAL

AUTOR: del Río Ortuño, Ramón

TUTOR: Esteso Álvarez, Ana

COTUTORA: Díaz Alemany, Mareva

CURSO ACADÉMICO: 2023/24

ÍNDICE DEL PRESUPUESTO

Contenido

1. Introducción	3
2. Descripción de la Problemática	3
3. Diseño, implementación y aplicación.....	4
4. Aplicación a un caso real	4
5. Realización y redacción del TFG	5
6. Resumen.....	5

Índice de tablas

Tabla 1. Presupuesto asociado al Bloque 1. Descripción de la Problemática. Fuente: Elaboración propia.	3
Tabla 2. Presupuesto asociado al Bloque 2. Diseño, implementación y aplicación. Fuente: Elaboración propia.	4
Tabla 3. Presupuesto asociado al Bloque 3. Aplicación a un caso real. Fuente: Elaboración propia.	4
Tabla 4. Presupuesto asociado al Bloque 5. Realización y redacción del TFG. Fuente: Elaboración propia.	5
Tabla 5. Resumen del Presupuesto global de la realización del proyecto. Fuente: Elaboración propia.	5

1. Introducción

Como última parte del TFG se busca la estimación monetaria de lo que supone la realización de este trabajo. En este presupuesto se incluye la ejecución y elaboración de las diferentes propuestas relacionadas con las herramientas heurísticas relacionadas con la programación de exámenes. Para desglosar los costos imputados en el trabajo se ha decidido dividirlo en tres partes diferenciadas teniendo en cuenta los capítulos que se incluyen en el presente documento de investigación:

- Bloque 1. Descripción de la Problemática. “Capítulo 2. DESCRIPCIÓN DE LA PROBLEMÁTICA” y “Capítulo 3. ANTECEDENTES”.
- Bloque 2. Diseño, implementación y aplicación. “Capítulo 4. HERRAMIENTA HEURÍSTICA CONSTRUCTIVA PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS” y “Capítulo 5. HERRAMIENTA HEURÍSTICA DE BÚSQUEDA LOCAL PARA LA PROGRAMACIÓN DEL CALENDARIO DE EXÁMENES CONSIDERANDO LA DISTANCIA ENTRE AULAS”.
- Bloque 3. Aplicación a un caso real. “Capítulo 6. APLICACIÓN A LA ETSII. ANÁLISIS DE RESULTADOS”.
- Bloque 4. Realización y redacción del TFG.

2. Descripción de la Problemática

En este apartado se detallan los costes relativos a la búsqueda y consesuación en relación a la información previa a la realización de las herramientas heurísticas:

OPERACIONES	DESCRIPCIÓN DE LA OPERACIÓN	TIEMPO EMPLEADO (h)	TASA HORARIA (€/h)	IMPORTE (€)
1. Búsqueda y comprensión de información relativa la problemática	Entendimiento y búsqueda de la información relativa a la información previa a las herramientas destacando la documentación relativa a la ETSII y la problemática, contando los antecedentes de investigación.	25h	15€/h	375 €
2. Búsqueda y comprensión de información relativa a las heurísticas	Entendimiento y búsqueda de la información relativa a la información de las herramientas heurísticas tanto como conocimiento previo y aplicadas al ETP.	53h	15€/h	795 €
3. Tutorías	Resolución de dudas	4h	50€/h	200 €
TOTAL				1.370 €

Tabla 1. Presupuesto asociado al Bloque 1. Descripción de la Problemática (Fuente: Elaboración propia)

3. Diseño, implementación y aplicación

En segundo lugar, se abordan los costes empleados en la realización de las diferentes heurísticas para la optimización de la problemática de la programación de exámenes, contando también su desarrollo EN la herramienta de implementación utilizada para la realización de este:

OPERACIONES	DESCRIPCIÓN DE LA OPERACIÓN	TIEMPO EMPLEADO (h)	TASA HORARIA (€/h)	IMPORTE (€)
1. Formación relativa a MS Excel Visual Basic	Asimilar y comprender los conocimientos relativos a la programación en el lenguaje de la herramienta de programación Visual Basic	20h	15€/h	300 €
2. Diseño de los modelos heurísticos	Debate y decisión sobre la base de datos y la progresión de las heurísticas	64h	15€/h	960 €
3. Desarrollo y programación de las heurísticas	Realización del código y del documento Excel	78h	15€/h	1.170 €
4. Implementación de la herramienta	Implementación de la herramienta	12h	15€/h	180 €
3. Tutorías	Resolución de dudas	12h	50€/h	600 €
TOTAL				3.210 €

Tabla 2. Presupuesto asociado al Bloque 2. Diseño, implementación y aplicación (Fuente: Elaboración propia)

4. Aplicación a un caso real

En este tercer bloque se abordan los costes incurridos en la puesta en escena para el caso particular de la ETSII y sus recursos.

OPERACIONES	DESCRIPCIÓN DE LA OPERACIÓN	TIEMPO EMPLEADO (h)	TASA HORARIA (€/h)	IMPORTE (€)
1. Aplicación a un caso real	Aplicación a la ETSII de la heurística constructiva diseñada y los Algoritmos de búsqueda local.	35h	15€/h	525 €
2. Implementación de la herramienta en un caso real	Utilización del programa para el caso de la ETSII.	14h	15€/h	210 €
3. Tutorías	Resolución de dudas	10h	50€/h	500 €
TOTAL				1.235 €

Tabla 3. Presupuesto asociado al Bloque 3. Aplicación a un caso real (Fuente: Elaboración propia)

5. Realización y redacción del TFG

Por último, se tratan los costes relacionados con el trabajo realizado para la redacción del documento.

OPERACIONES	DESCRIPCIÓN DE LA OPERACIÓN	TIEMPO EMPLEADO (h)	TASA HORARIA (€/h)	IMPORTE (€)
1. Redacción del TFG	Realización de los diferentes capítulos del documento.	90h	15€/h	1.350 €
2. Revisión del TFG	Revisión de los realizado para asegurar la correcta redacción del documento.	10h	15€/h	150 €
3. Tutorías	Resolución de dudas	5h	50€/h	250 €
TOTAL				1.750 €

Tabla 4. Presupuesto asociado al Bloque 5. Realización y redacción del TFG (Fuente: Elaboración propia)

6. Resumen

Para concluir, se incluye un resumen breve de los gastos del proyecto que se derivan de los subapartados anteriores. Adicionalmente, se calcula que un porcentaje del 5% de los gastos totales está relacionado con los materiales necesarios para completar el proyecto, incluidos los materiales de oficina, impresos, transporte y computadoras.

OPERACIONES	IMPORTE (€)
BLOQUE 1. Descripción de la Problemática	1.370 €
BLOQUE 2. Diseño, implementación y aplicación	3.210 €
BLOQUE 3. Aplicación a un caso real	1.235 €
BLOQUE 4. Realización y redacción del TFG	1.750 €
TOTAL	7.565 €

Tabla 5. Resumen del Presupuesto global de la realización del proyecto (Fuente: Elaboración propia)