

Estructuras de Datos y Algoritmos

Presentación de la asignatura

☹ All wrongs reversed – bajo licencia [CC-BY-NC-SA 4.0](#)



Universidad
Zaragoza

Dpto. de Informática e Ingeniería de Sistemas
Universidad de Zaragoza, España

Grado en Ingeniería Informática
UNIVERSIDAD DE ZARAGOZA



Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía
- 7 Tareas

Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía
- 7 Tareas

Profesorado

Coordinador de la asignatura

- Ricardo J. Rodríguez (rjrodriguez@unizar.es)

Profesores de teoría y problemas



Gonzalo Esteban
(mañanas, 421 y 423)
g.esteban@unizar.es



Ricardo J. Rodríguez
(mañanas, 423)
rjrodriguez@unizar.es



Yolanda Villate
(tardes)
yvillate@unizar.es

Gonzalo Esteban

- Despacho AYD (1ª planta Edificio Ada Byron)
- **Tutorías:** L y J de 10:00 a 12:00, X de 12:00 a 14:00
 - Reservar con un mínimo de 24 horas de antelación
 - Posibles cambios puntuales (mirar el calendario de citas)
 - **Reservas a través de Google Calendar** (enlace también disponible en Moodle)
 - Indicar en la reserva: *asignatura, tipo de tutoría*
 - Para otros horarios, consultar vía correo electrónico

<https://webdiis.unizar.es/asignaturas/EDA/>

Profesorado

Información de interés para grupo de mañanas (423)

Gonzalo Esteban

- (consultar detalles en diapositiva anterior)

Ricardo J. Rodríguez

- Despacho D0.05
- **Tutorías:** M de 12:00 a 14:00 y X de 12:00 a 14:00
 - Reservar con un mínimo de 24 horas de antelación
 - Posibles cambios puntuales (mirar el calendario de citas)
 - [Reservas a través de Google Calendar](#) (enlace también disponible en Moodle)
 - Indicar en la reserva: *asignatura, tema a tratar, presencial/remoto*
 - Para otros horarios, consultar vía correo electrónico

<https://webdiis.unizar.es/asignaturas/EDA/>

Profesorado

Profesores de prácticas



Jordi Bernad
jbernad@unizar.es



Javier Carrillo-Mondéjar
jcarrillo@unizar.es



Alfonso López
alfonso.lopezr@unizar.es



Yolanda Villate

Índice

- 1 Profesorado
- 2 Descripción de la asignatura**
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía
- 7 Tareas

Descripción de la asignatura

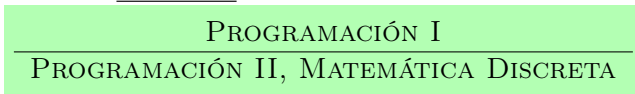
ESTRUCTURAS DE DATOS Y ALGORITMOS

- Asignatura *obligatoria* de 6 créditos ECTS
- Formación previa en programación (necesario) y matemática (recomendable)
 - PROGRAMACIÓN II
 - MATEMÁTICA DISCRETA
- **Objetivo principal**
 - Identificar, diseñar y definir Tipos Abstractos de Datos, independientemente de su implementación
- Enlace a curso en Moodle:
<https://moodle.unizar.es/add/course/view.php?id=165465>
 - Clave de auto-matrícula: *****

Descripción de la asignatura

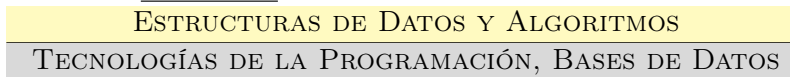
Relación con otras asignaturas de la carrera

Primer curso

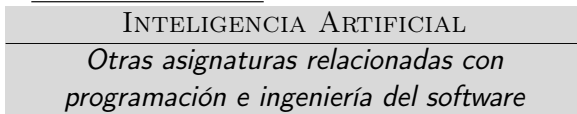


- Diseño imperativo descendente
- Diseño recursivo
- Estructuras de datos sencillas
- Análisis de la eficiencia de algoritmos
- Grafos (conceptos, representación)

Segundo curso



Tercer curso y siguientes



Descripción de la asignatura

Resultados de aprendizaje

- Definir tipos abstractos de datos (TAD's) independientemente de su implementación.
- Implementar TAD's en un lenguaje de programación modular.
- Implementar y utilizar algunos TAD's fundamentales, como pilas, colas, listas, árboles de búsqueda, tablas hash y grafos.
- Comparar distintas alternativas de implementación de TAD's con respecto al tiempo de ejecución de algoritmos y al uso de la memoria.
- Afrontar el diseño modular de programas de tamaño medio identificando, definiendo e implementando los TAD's necesarios.
- *Aplicar los esquemas algorítmicos básicos (como dividir para vencer, búsqueda con retroceso, voracidad...) a la resolución de problemas.*

Más información: *Guía docente de la asignatura, curso*

Descripción de la asignatura

Programa

T1 Programación con Tipos Abstractos de Datos

T2 Tipos de datos lineales

T3 Tipos de datos arborescentes

T4 Tipos de datos funcionales

T5 Introducción a los grafos

T6 Introducción a los esquemas algorítmicos

Descripción de la asignatura

Metodología

Clases de teoría

- Aprendizaje de conceptos
- 2 horas semanales
- Asistencia voluntaria (**pero muy recomendable**)

Clases de problemas

- Aplicación de conceptos
- 1 hora semanal
- Grupos desdoblados
- Asistencia voluntaria (**pero muy recomendable**)

Descripción de la asignatura

Metodología

Clases prácticas

- 5 grupos de prácticas
- 2 horas en semanas alternas

Descripción de la asignatura

Prácticas

5 prácticas, distribuidas en 6 sesiones:

- P1** Implementación de un TAD en memoria estática
- P2** Implementación de un TAD genérico en memoria estática
- P3** Implementación de TADs lineales en memoria dinámica (dos sesiones)
- P4** Implementación de TADs arborescentes en memoria dinámica
- P5** Diseño de programas basado en TADs

Descripción de la asignatura

Prácticas – grupo 421

- **Elegid grupo de prácticas preferentemente de los previstos para el grupo de mañanas correspondiente**
- Reserva plaza en [tarea disponible en Moodle](#)
- Plazo límite: **14 de septiembre (a las 10:05)**
- Horarios (grupo 421):
 - **Lunes semana A, de 08:00 a 10:00, Laboratorio L0.04**
 - **Lunes semana B, de 08:00 a 10:00, Laboratorio L0.04**
 - **Miércoles semana A, de 12:00 a 14:00, Laboratorio L0.02**
 - **Miércoles semana B, de 12:00 a 14:00, Sala de Informática 10 (Edificio Torres Quevedo)**
 - **Jueves semana B, de 12:00 a 14:00, Laboratorio L0.04**

Descripción de la asignatura

Prácticas – grupo 423

- **Elegid grupo de prácticas preferentemente de los previstos para el grupo de mañanas correspondiente**
- Reserva plaza en [tarea disponible en Moodle](#)
- Plazo límite: **14 de septiembre (a las 10:05)**
- Horarios (grupo 423):
 - ~~Miércoles semana B, de 08:00 a 10:00, Laboratorio L0.02~~ (grupo cerrado)
 - **Jueves semana A, de 10:00 a 12:00, Laboratorio L0.04**
 - **Jueves semana B, de 10:00 a 12:00, Laboratorio L0.04**
 - **Viernes semana A, de 08:00 a 10:00, Laboratorio L0.04**
 - **Viernes semana B, de 08:00 a 10:00, Laboratorio L0.04**

Descripción de la asignatura

Prácticas

- P1 y P2 son **individuales**
- P3 a P5 por **parejas (del mismo grupo de laboratorio)**
 - **Tenéis que formar esas parejas**
 - Reserva en [tarea disponible en Moodle](#)
 - Plazo límite: **9 de octubre (a las 19:05)**
 - Tras el plazo, **se formarán equipos entre aquellos que no estén inscritos en ninguna pareja**
 - [Foro “Buscando pareja de prácticas” en Moodle](#)
- Cada pareja de prácticas tendrá un profesor tutor
- Lenguaje: C++
- Computador: lab000 (mediante IDE o línea de comandos)
 - Puede hacerse en otra máquina, **pero deberá probarse finalmente en lab000, que es donde se evaluará**
 - Más información: documento [“Realización y entrega de prácticas en las máquinas del DIIS”](#)

Descripción de la asignatura

Prácticas

■ Hay 2 entregas de prácticas

- Se indicará en los enunciados
 - **Para su realización se reutilizarán código de prácticas anteriores**
- El código resultante de las prácticas deberá:
- Estar documentado
 - Ser legible
 - Utilizar diseño descendente
 - Ser ejecutable sin errores
 - Mostrar un funcionamiento conforme a lo solicitado

Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso**
- 4 Evaluación
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía
- 7 Tareas

Cómo trabajaremos: rutinas del curso

- El diseño de las clases ataca las **familias de error** que se repiten, curso tras curso, en el examen
- Cuatro **rutinas permanentes** (las veréis una y otra vez):
 - 1 **Regla QUÉ/CÓMO** en cada especificación (desde la semana 1)
 - 2 **Tabla de decisión** estructura $\leftrightarrow$ requisito (arranca en la lección 5 y se cierra en la 19)
 - 3 **Método de análisis de coste en 6 pasos**, al cerrar cada implementación (desde la lección 8)
 - 4 **Galería de antipatrones**: un error clásico por estructura
- En las clases de problemas: calentamiento con un antipatrón, resolución guiada, y cierre con análisis de coste y tabla de decisión

Cómo trabajaremos: rutinas del curso

Las familias de error

Cada rutina, ejercicio o aviso llevará la insignia de su familia:

- **F1** frontera abstracción/implementación: fugas de implementación en la especificación; operaciones *parciales* sin dominio; restricciones omitidas
- **F2** estructuras enlazadas: duplicar en vez de enlazar; punteros recíprocos que faltan; cardinales sin mantener
- **F3** elección de estructura: lista donde se pide eficiencia; mal uso del montículo; dispersión sin justificar
- **F4** análisis de coste: sub-operaciones sin sumar; promedio $\neq$ peor caso; variables sin definir
- **EV** evaluación formativa: checklist «antes de entregar» y co-corrección de ejercicios de exámenes anteriores

Salen de las cajas de “Errores frecuentes” de los exámenes de cursos anteriores: son, literalmente, **los errores que cuestan puntos**

Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación**
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía
- 7 Tareas

Evaluación de las prácticas

- Las entregas de prácticas se compilarán y ejecutarán en el lab000
- Evaluadas de acuerdo a su comportamiento en esa máquina y a las instrucciones específicas dadas en los enunciados
- Cada entrega tendrá un plazo límite y unas instrucciones para la entrega
 - **No se admitirán entregas fuera de plazo**
 - **Ni entregas por ninguna otra vía que la indicada en las instrucciones**
- Cada entrega se calificará con una nota cuantitativa de 0 a 10
- Aspectos a valorar:
 - Funcionamiento de los programas según especificaciones
 - Calidad de su diseño y su presentación
 - Adecuada aplicación de los métodos de resolución
 - Capacidad de los integrantes del equipo para explicar y justificar el diseño realizado, si se les requiere

Evaluación de las prácticas

- Las entregas de prácticas se compilarán y ejecutarán en el lab000
- Evaluadas de acuerdo a su comportamiento en esa máquina y a las instrucciones específicas dadas en los enunciados
- Cada entrega tendrá un plazo límite y unas instrucciones para la entrega
 - **No se admitirán entregas fuera de plazo**
 - **Ni entregas por ninguna otra vía que la indicada en las instrucciones**
- Cada entrega se calificará con una nota cuantitativa de 0 a 10
- Aspectos a valorar:
 - Funcionamiento de los programas según especificaciones
 - Calidad de su diseño y su presentación
 - Adecuada aplicación de los métodos de resolución
 - Capacidad de los integrantes del equipo para explicar y justificar el diseño realizado, si se les requiere

IMPORTANTE – PRÁCTICA SUPERADA EN CURSOS ANTERIORES

- *Si la nota final de prácticas en curso 24-25 o 25-26 es ≥ 7.0 , se guarda para este curso*
- Se aplican condiciones:
 - **No** se tiene que apuntar a ningún grupo de prácticas
 - **No** se tiene que apuntar con ninguna pareja de prácticas

Evaluación global

- Examen escrito **(70 % de la nota final)**
 - Sin apuntes
 - Problemas de programación, preguntas conceptuales o resolver algún ejercicio
 - **Calificación mínima exigida de 5.0**
- Examen práctico de programación, en laboratorio **(30 % de la nota final)**
 - Ejercicios de programación similares a los realizados en prácticas o en clase
 - **Calificación mínima exigida de 5.0**
 - Aquellos que obtengan una nota final mínima de 5.0 en sus entregas de prácticas estarán exentos de realizar este examen práctico
 - Prevalece la nota obtenida en el examen práctico individual

Convocatorias

- **Primera:** miércoles 13 de enero, 2027 (14:00–20:00)
- **Segunda:** martes 2 de febrero, 2027 (14:00–20:00)

Evaluación global

Plagios

- Plagio total o parcial en las prácticas → **fraude académico**
 - Art. 30 «Prácticas irregulares en Reglamento de Normas de Evaluación del Aprendizaje las pruebas de evaluación distintas de fraude académico»,
 - Art. 31 «Faltas disciplinarias», [Reglamento de Normas de Convivencia](#).

Evaluación global

```
nota_ponderada := nota_exam_escrito*0.7 + nota_prácticas*0.3
si nota_exam_escrito >= 5.0 and nota_prácticas >= 5.0 entonces
    nota_final := nota_ponderada
sino {suspenseo seguro}
    nota_final := mínimo(4.0, nota_ponderada)
fsi
```

- Calificaciones obtenidas en las dos partes (examen escrito y prácticas) en primera convocatoria **se guardan para la segunda convocatoria del mismo curso académico**

Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación
- 5 Carga de trabajo para el estudiante**
- 6 Bibliografía
- 7 Tareas

Carga de trabajo para el estudiante

- Asignatura de 6 ECTS $\rightarrow \sim 150$ horas (en promedio) efectivas de trabajo
 - **60 horas de actividades presenciales activas**
 - 30 horas de clases de teoría
 - 15 horas de clases de problemas
 - 15 horas de clases prácticas
 - **25 horas de trabajo de programación para completar las tareas prácticas**
 - **60 horas de estudio personal efectivo**
 - Estudio de apuntes y textos
 - Resolución de problemas
 - Preparación clases y prácticas
 - Desarrollo de programas
 - **5 horas de pruebas de evaluación**

Carga de trabajo para el estudiante

Recordatorio de tiempos

- Un curso académico son 60 ECTS
- 1 ECTS, en media, son 25 horas de trabajo
 - 1 curso académico equivale a 1500 horas de trabajo
 - 2 meses de vacaciones → 150 horas de trabajo al mes

Cada semana supone 37.5 horas de trabajo

Descripción de la asignatura

¿Cómo puedo facilitar mi aprendizaje?

■ Asistir a clase

- Habiendo realizado el trabajo previo que se haya encargado
- Atendiendo y participando activamente en la clase
- Repasando después y comprendiendo cada lección

■ Resolver los problemas propuestos en las clases de problemas

■ Resolver y repasar los problemas propuestos en las prácticas

- Lápiz y papel
- Programación en computador
- Validación del código desarrollado

■ Discutir ideas con otros compañeros (ojo, no compartir código de prácticas)

■ Consultar dudas a los profesores en sus horarios de tutorías

Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía**
- 7 Tareas

Bibliografía

<https://psfunizar10.unizar.es/br13/egAsignaturas.php?id=8027>

Bibliografía básica

- *Data Structures and Algorithm Analysis in C++*, M.A. Weiss, 4th Edition, Pearson/Addison Wesley, 2013.
- *Fundamentos de Estructuras de Datos. Soluciones en Ada, Java y C++*, Z.J. Hernández y otros, Thomson, 2005.
- *Data Structures and Algorithm Analysis in C++*, C.A. Shaffer, 3rd Edition, Dover Publications, 2011.
- *Apuntes de Estructuras de datos y algoritmos*, J. Campos, 2da ed., v4, 2022.

C++

- *The C++ Programming Language*, B. Stroustrup, 4th Edition, Addison-Wesley, 2013.

Bibliografía

<https://psfunizar10.unizar.es/br13/egAsignaturas.php?id=8027>

Ejercicios

- *Estructura de datos y algoritmos. Ejercicios y problemas resueltos*, N. Martí Oliet, Y. Ortega Mallén, J.A. Verdejo López, Pearson Prentice Hall, 2003.
- *Estructura de datos. Libro de problemas*, L. Joyanes, I. Zahonero, M. Fernández y L. Sánchez, McGraw Hill, 1999.

Bibliografía complementaria

- *Estructuras de datos. Especificación, diseño e implementación*, X. Franch, 3a edición, Ediciones UPC, 2001.
- *Handbook of Data Structures and Applications*, D.P. Mehta y S. Sahni, Chapman & Hall/CRC, 2005

Índice

- 1 Profesorado
- 2 Descripción de la asignatura
- 3 Cómo trabajaremos: rutinas del curso
- 4 Evaluación
- 5 Carga de trabajo para el estudiante
- 6 Bibliografía
- 7 Tareas**

Para esta semana

1 Verifica si tu grupo de teoría es correcto en Moodle

- En Moodle, entra en la asignatura y ve a Participantes
- Busca tu nombre en la lista y comprueba tu **Grupo**

2 Escoge un grupo de prácticas de los reservados a vuestro grupo de mañanas (preferiblemente)

- Elección abierta hasta el 14 de septiembre (a las 10:05)

3 Busca una pareja de prácticas

- Usa el foro “Buscando pareja de prácticas” en Moodle si lo necesitas
- Tanto tú como tu pareja de prácticas debéis estar en el mismo grupo

Para esta semana

1 Verifica si tu grupo de teoría es correcto en Moodle

- En Moodle, entra en la asignatura y ve a **Participantes**
- Busca tu nombre en la lista y comprueba tu **Grupo**

2 Escoge un grupo de prácticas de los reservados a vuestro grupo de mañanas (preferiblemente)

- Elección abierta hasta el 14 de septiembre (a las 10:05)

3 Busca una pareja de prácticas

- Usa el foro “Buscando pareja de prácticas” en Moodle si lo necesitas
- Tanto tú como tu pareja de prácticas debéis estar en el mismo grupo

Para la próxima semana

1 Insíbete en un subgrupo (equipo) de prácticas

- Tanto tú como tu pareja de prácticas debéis inscribiros en el mismo subgrupo (equipo)
- Elección abierta hasta el 9 de octubre (a las 19:05)

Estructuras de Datos y Algoritmos

Presentación de la asignatura

☹ All wrongs reversed – bajo licencia [CC-BY-NC-SA 4.0](#)



Universidad
Zaragoza

Dpto. de Informática e Ingeniería de Sistemas
Universidad de Zaragoza, España

Grado en Ingeniería Informática
UNIVERSIDAD DE ZARAGOZA

