

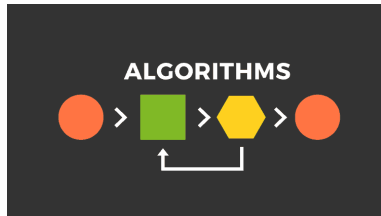
# Algoritmos y Estructuras de Datos

## Tema 0: Introducción

Grado Imat. Escuela ICAI

Juan C. Aguí García

January 2024



# Información General (I)

- **Horario (Salvo cambios puntuales)**

- Martes 8:00 am. Clases Prácticas
- Miércoles 8:00 am. Clases Teóricas

- **Horario (Salvo cambios puntuales)**

- Martes 8:00 am. Clases Prácticas
- Miércoles 8:00 am. Clases Teóricas

- **Profesores**

- Juan Claudio Agüi García (jcagui@icai.comillas.edu)
- Sheila De la Morena (sdemorena@icai.comillas.edu)

- **Horario (Salvo cambios puntuales)**
  - Martes 8:00 am. Clases Prácticas
  - Miércoles 8:00 am. Clases Teóricas
- Profesores
  - Juan Claudio Agüi García (jcagui@icai.comillas.edu)
  - Sheila De la Morena (sdemorena@icai.comillas.edu)
- Créditos: 6

- **Horario (Salvo cambios puntuales)**
  - Martes 8:00 am. Clases Prácticas
  - Miércoles 8:00 am. Clases Teóricas
- Profesores
  - Juan Claudio Agüi García (jcagui@icai.comillas.edu)
  - Sheila De la Morena (sdemorena@icai.comillas.edu)
- Créditos: 6
- Portal Moodle
  - Documentación, teoría y Practicas
  - Entregas y calificaciones

- **Horario (Salvo cambios puntuales)**
  - Martes 8:00 am. Clases Prácticas
  - Miércoles 8:00 am. Clases Teóricas
- Profesores
  - Juan Claudio Agüi García (jcagui@icai.comillas.edu)
  - Sheila De la Morena (sdemorena@icai.comillas.edu)
- Créditos: 6
- Portal Moodle
  - Documentación, teoría y Practicas
  - Entregas y calificaciones
- Teams ( Cód acceso: **whshu09** )
  - Notas, avisos, comentarios, etc...
  - Aportes y materiales complementarios

- **Horario (Salvo cambios puntuales)**
  - Martes 8:00 am. Clases Prácticas
  - Miércoles 8:00 am. Clases Teóricas
- Profesores
  - Juan Claudio Agüi García (jcagui@icai.comillas.edu)
  - Sheila De la Morena (sdemorena@icai.comillas.edu)
- Créditos: 6
- Portal Moodle
  - Documentación, teoría y Practicas
  - Entregas y calificaciones
- Teams ( Cód acceso: **whshu09** )
  - Notas, avisos, comentarios, etc...
  - Aportes y materiales complementarios
- github site : [https://github.com/juanclaudioagui/Estructuras\\_algoritmos\\_ICAI24\\_Alumnos](https://github.com/juanclaudioagui/Estructuras_algoritmos_ICAI24_Alumnos)  
fragmentos de código.



# Objetivos del curso (1)

- 1 Comprender en profundidad el concepto de Algoritmo  
Tipos básicos, Especificación y Análisis de complejidad

# Objetivos del curso (1)

- 1 Comprender en profundidad el concepto de Algoritmo  
Tipos básicos, Especificación y Análisis de complejidad
- 2 Conocer las estructuras básicas de Datos que soportan los algoritmos  
Colecciones, Listas enlazadas, Árboles

# Objetivos del curso (1)

- 1 Comprender en profundidad el concepto de Algoritmo  
Tipos básicos, Especificación y Análisis de complejidad
- 2 Conocer las estructuras básicas de Datos que soportan los algoritmos  
Colecciones, Listas enlazadas, Árboles
- 3 Conocer los algoritmos básicos de Ordenación  
Entendiendo el papel de la ordenación, y sus tipos básicos y complejidad

# Objetivos del curso (1)

- 1 Comprender en profundidad el concepto de Algoritmo  
Tipos básicos, Especificación y Análisis de complejidad
- 2 Conocer las estructuras básicas de Datos que soportan los algoritmos  
Colecciones, Listas enlazadas, Árboles
- 3 Conocer los algoritmos básicos de Ordenación  
Entendiendo el papel de la ordenación, y sus tipos básicos y complejidad
- 4 Elementos básicos del algoritmo de búsqueda  
Basados en árboles

# Objetivos del curso (1)

- 1 Comprender en profundidad el concepto de Algoritmo  
Tipos básicos, Especificación y Análisis de complejidad
- 2 Conocer las estructuras básicas de Datos que soportan los algoritmos  
Colecciones, Listas enlazadas, Árboles
- 3 Conocer los algoritmos básicos de Ordenación  
Entendiendo el papel de la ordenación, y sus tipos básicos y complejidad
- 4 Elementos básicos del algoritmo de búsqueda  
Basados en árboles
- 5 Nociones sobre algoritmos avanzados  
Greedy, Programación Dinámica, MonteCarlo, etc...

# Objetivos del curso(2)

Al final del curso el alumno debería estar capacitado para:

- Entender la literatura profesional sobre la materia, seleccionar (si existe) el algoritmo más apropiado para resolver un problema concreto

# Objetivos del curso(2)

Al final del curso el alumno debería estar capacitado para:

- Entender la literatura profesional sobre la materia, seleccionar (si existe) el algoritmo más apropiado para resolver un problema concreto
- realizar buenos diseños de algoritmos específicos, cumpliendo especificaciones

# Objetivos del curso(2)

Al final del curso el alumno debería estar capacitado para:

- Entender la literatura profesional sobre la materia, seleccionar (si existe) el algoritmo más apropiado para resolver un problema concreto
- realizar buenos diseños de algoritmos específicos, cumpliendo especificaciones



# Objetivos del curso(2)

Al final del curso el alumno debería estar capacitado para:

- Entender la literatura profesional sobre la materia, seleccionar (si existe) el algoritmo más apropiado para resolver un problema concreto
- realizar buenos diseños de algoritmos específicos, cumpliendo especificaciones

Además, debería conseguir mejorar el estilo y la estructura de sus programas, así como avanzar en las técnicas de programación y depuración.

# De qué va ésto ??

# De qué va ésto ??

## Gestionando Datos de forma eficiente

- Grandes Cantidades de Datos
- Frecuentemente Homogéneos, pero no siempre
- cómo los almacenamos, dependerá del uso que queramos hacer de ellos
- Interfaz básica de las **Colecciones**: Añadir, eliminar, insertar, primero, último, etc...

# De qué va ésto ??

## Gestionando Datos de forma eficiente

- Grandes Cantidades de Datos
- Frecuentemente Homogéneos, pero no siempre
- cómo los almacenamos, dependerá del uso que queramos hacer de ellos
- Interfaz básica de las **Colecciones**: Añadir, eliminar, insertar, primero, último, etc...

## Operaciones Básicas sobre los datos

- **Búsqueda** (*Search*)
- **Ordenación** (*Sort*)



Durante el transcurso de las clases se combinará:

- Lecciones teoría<sup>1</sup>
- Ejercicios prácticos para asegurar una correcta comprensión de la misma. **Las prácticas se harán en grupos de 3 o menos estudiantes.**
- MidTerm Challenge
- Proyecto Final

---

<sup>1</sup>Bueno, no tanto... 😊

# Enlazando la teoría y la práctica

- a al final de la clase de teoría (Miércoles) **se libera la practica asociada con un breve comentario por parte del profesor**
- b Al principio de la clase de prácticas (Lunes siguiente) **Uno o más de los equipos habrá de salir y compartir su aproximación al problema**
- c Trabajo en clase con soporte del/los profesores
- d **Entrega** de la práctica el domingo tras el lunes de trabajo en clase.  
*Miercoles( Teoria) → Finde → Lunes(Practica) → Domingo(Entrega)*

- Tema 0: Presentación y planificación de la asignatura
- Tema 1: Introducción al desarrollo de algoritmos
- Tema 2: Algoritmos básicos
- Tema 3: Complejidad algorítmica
- Tema 4: Estructuras de Datos básicas:
  - Colecciones
  - Programación Orientada a Objetos
  - Listas Enlazadas
  - Pilas y Colas
- Tema 5: Algoritmos Básicos de Ordenación
- Tema 6: Algoritmos de Búsqueda
  - Estructuras de Árboles
  - Ordenación y Búsqueda en Arboles
- Tema 7: Algoritmos avanzados: Greedy, Grafos, Dynamic programming.
- **Proyecto Final**

# Prerequisitos para el curso



## Aprende a leer y escribir Pseudocódigo

Todo los lenguajes son iguales... o diferentes, pero el Pseudocódigo es uno!!

## Aprende a leer y escribir Pseudocódigo

Todo los lenguajes son iguales... o diferentes, pero el Pseudocódigo es uno!!

## Un buen nivel de Python

Hazte con un libro de Python y estudial oen el metro, o paseando !  
ver las referencias bibliográficas

## Aprende a leer y escribir Pseudocódigo

Todo los lenguajes son iguales... o diferentes, pero el Pseudocódigo es uno!!

## Un buen nivel de Python

Hazte con un libro de Python y estudial oen el metro, o paseando !  
ver las referencias bibliográficas

## Know your tools: VSCode o algo similar

Aprende el Debugger (es clave !!)

## Aprende a leer y escribir Pseudocódigo

Todo los lenguajes son iguales... o diferentes, pero el Pseudocódigo es uno!!

## Un buen nivel de Python

Hazte con un libro de Python y estudial oen el metro, o paseando !  
ver las referencias bibliográficas

## Know your tools: VSCode o algo similar

Aprende el Debugger (es clave !!)

## Practice Makes perfect

Sólo implementando en código los algoritmos llegarás a entenderlos!

## La asistencia a clase es obligatoria según las normas de la ETSI-ICAI

Registro en BIDI al principio de cada clase

### ● Convocatoria Ordinaria:

- Exámenes (60% del total):
  - 20% Examen intercuatrimestral: Teoría
  - 40% Examen final: Teoría

La nota mínima del examen de Teoría final es de 5 para poder hacer media con el resto de exámenes y prácticas.

- Evaluación del rendimiento (40% del total):
  - 40% Informes de prácticas, ejercicios adicionales, explicación en clase y entregables

La nota mínima de la parte práctica es de 4 para poder hacer media con el resto de exámenes.

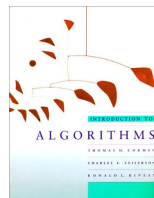
### ● Convocatoria Extraordinaria:

- Examen (85% del total):

La nota mínima del examen de Teoría extraordinario para poder hacer media con el resto de notas es de 4.

## ● Algoritmos

- 1 Introduction to Algorithms, T.H. Cormen, C.E. Leiserson, R.L. Rivest y C. Stein, The MIT Press, 4th Ed., 2022
- 2 Algorithms, R. Sedgewick, Segunda edición, Addison Wesley, 1988.
- 3 Algorítmica, J. Galve, J. C. González, A. Sánchez, J. A. Velásquez, Editorial RA-MA, 1993



## ● Programación en Python<sup>a</sup>

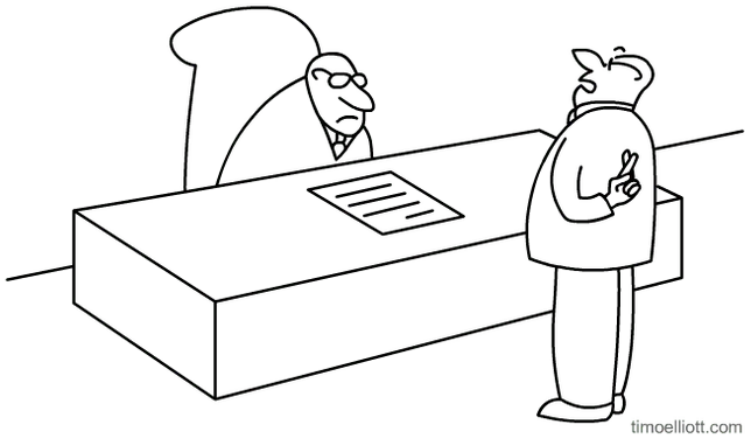
- 1 PEP8. Ver <https://pep8.org/> Guía de estilo
- 2 Think Python, by Allen B. Downey. See <https://greenteapress.com/wp/think-python-2e/>



<sup>a</sup>Documentos de libre acceso



Eof Tema 0: Gracias !



*"Yes sir, you can absolutely trust those numbers"*

