

# UD3: Estructuras de control de flujo. Manejo de Excepciones.

JOSÉ CLIMENT PENADÉS

# Índice

- ▶ Ubicación y justificación
- ▶ Normativa
- ▶ Objetivos
- ▶ Competencias
- ▶ Contenidos
- ▶ Metodología
- ▶ Agrupamientos
- ▶ Sesiones
- ▶ Evaluación
- ▶ Atención a la diversidad
- ▶ Recursos y bibliografía
- ▶ Actividades

# Ubicación y justificación

- ▶ Tercera unidad del modulo.
- ▶ 1º Evaluación.
- ▶ 25 horas = 12'5 días (2 horas por día)
- ▶ RA 3: Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
- ▶ Software propio industrial.
- ▶ Necesidad de crear, actualizar y mantener software

# Normativa

- ▶ Ley Orgánica de educación (2/2006)
- ▶ Ley Orgánica de Modificación de la LOE (3/2020)
- ▶ Ley Orgánica de la Formación Profesional (3/2022)
- ▶ Real decreto del calendario de implantación (278/2023)
- ▶ Instrucciones para el alumnado procedente de otros países(16/07/2019)
- ▶ Orden sobre la prevención de riesgos laborales (1/2014)
- ▶ Instrucciones de inicio de curso (08/08/2024)
- ▶ Real decreto de ordenación del Sistema de Formación Profesional.(659/2023)
- ▶ Título del CFGS DAM (450/2010)
- ▶ Actualización del CFGS DAM(405/2023)
- ▶ Modificación del CFGS DAM (500/2024)
- ▶ Currículo nacional del CFGS DAM (EDU/2000/2010)
- ▶ Currículo autonómico del CFGS DAM (58/2012)
- ▶ Decreto de equidad e inclusión(104/2018)
- ▶ Orden de respuesta educativa a la inclusión(20/2019)
- ▶ Decreto de igualdad y convivencia (195/2022)

# Objetivos

Objetivos	
OD1	Identificar y aplicar estructuras de selección adecuadamente.
OD2	Identificar y aplicar correctamente las estructuras de repetición.
OD3	Utilizar estructuras de salto para modificar el flujo de ejecución según las condiciones establecidas.
OD4	Diseñar algoritmos que integren condiciones compuestas y operadores lógicos.
OD5	Identificar y diferenciar las distintas técnicas para el manejo de excepciones.
OD6	Hacer uso de las distintas técnicas de manejo de excepciones.
OD7	Probar, depurar y corregir errores mediante técnicas de manejo de excepciones.
OD8	Representar y documentar el flujo lógico de un programa mediante diagramas de flujo y pseudocódigo.

# Competencias

- ▶ Competencias pps:
  - ▶ C3: Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.
  - ▶ C4: Mantener el espíritu de innovación y actualización en el ámbito de su trabajo para adaptarse a los cambios tecnológicos y organizativos de su entorno profesional.

# Contenidos

## 3. Uso de estructuras de control.

- 3.1. Estructuras de selección.
- 3.2. Estructuras de repetición.
- 3.3. Estructuras de salto.
- 3.4. Control de excepciones.
- 3.5 Aserciones.

- 3.6. Codificación, edición y compilación de programas con estructuras de control.
- 3.7. Prueba y depuración.
- 3.8. Documentación.

# Metodología

- ▶ Práctica y motivadora
- ▶ Orientaciones metodológicas:
  - ▶ Conocimientos previos del alumnado
  - ▶ Aprendizaje significativo y funcional
  - ▶ Metodologías activas



# Actividades E/A

- ▶ Actividades de presentación
- ▶ Actividades de evaluación de conocimientos previos
- ▶ Actividades de desarrollo de contenidos
- ▶ Actividades de cierre o síntesis-resumen
- ▶ Actividades de diagnóstico
- ▶ Actividades de refuerzo y ampliación
- ▶ Actividades evaluables o de evaluación

# Agrupamientos

- ▶ Trabajo individual
- ▶ Trabajo cooperativo en grupos pequeños
- ▶ Trabajos grupo-clase

# Sesiones

Planificación de las sesiones	
S1	- Presentación de las estructuras de control: definición y ejemplos teóricos (selección, repetición, salto).
S2 – S3	- Debate y análisis de condiciones y operadores lógicos en las estructuras de selección y repetición. - Ejercicios de pseudocódigo de selección y repetición.
S4 – S9	- Práctica en el uso de estructuras “if”, “switch”, “for” y “while”. - Ejercicios prácticos en grupo.
S10	- Introducción a estructuras de salto y su impacto en el flujo del programa. - Ejercicios prácticos sobre estructuras.
S11 – S15	- Explicación y ejemplos sobre el manejo de excepciones. - Ejercicios sobre el manejo de excepciones en el IDE.

# Sesiones

S16 S17	-	<ul style="list-style-type: none"><li>- Explicación sobre la depuración y las funciones integradas en el IDE.</li><li>- Actividades de depuración: identificar y corregir errores mediante el uso de excepciones.</li><li>- Revisión de casos reales.</li></ul>
S18 S19	-	<ul style="list-style-type: none"><li>- Trabajo en pequeños grupos para documentar y representar el flujo lógico de un programa con diagramas y pseudocódigo.</li></ul>
S20 S23	-	<ul style="list-style-type: none"><li>- Actividades de integración que combinen estructuras de repetición, salto, selección, control y manejo de excepciones, con tutorías y debate final.</li></ul>
S24 S25	-	<ul style="list-style-type: none"><li>- Actividades de ampliación y refuerzo.</li><li>- Cuestionario sobre los conceptos vistos durante la unidad.</li><li>- Resolución de dudas.</li></ul>

# Evaluación

## ▶ Proceso de enseñanza

- ▶ Autoevaluación de la metodología
- ▶ A mitad de la unidad y al final
- ▶ Rubrica de autoevaluación y cuaderno del docente

## ▶ Proceso de aprendizaje

- ▶ Criterios de evaluación del RA
- ▶ Evaluación inicial, continua y final
- ▶ Rubricas de evaluación
- ▶ Actividades y observación

# Evaluación

RA3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

- a) Se ha escrito y probado código que haga uso de estructuras de selección.
- b) Se han utilizado estructuras de repetición.
- c) Se han reconocido las posibilidades de las sentencias de salto.
- d) Se ha escrito código utilizando control de excepciones.
- e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- f) Se han probado y depurado los programas.
- g) Se ha comentado y documentado el código.
- h) Se han creado excepciones.
- i) Se han utilizado aserciones para la detección y corrección de errores durante la fase de desarrollo.

# Evaluación

- ▶ Actitud/trabajo diario 10%.
- ▶ Proyecto 30%.
- ▶ Prueba final 20%.
- ▶ Pruebas y actividades 40%.

# Atención a la diversidad

- ▶ Sesiones de refuerzo y ampliación
- ▶ Adaptaciones en acceso y tiempo según la normativa



# Recursos y bibliografía

- ▶ Usaremos el ordenador, proyector, pizarra, documentación y material de elaboración propia que el alumnado tendrá disponible en aulas.
- ▶ Además, usaremos un IDE, en este caso eclipse y un procesador de textos (LibreOffice Writer)

# Actividad 1

- ▶ **Actividad: Clasificación de productos según temporada agrícola**
- ▶ **Contexto laboral:**
  - ▶ Formas parte del equipo de desarrollo de una cooperativa agrícola de la huerta valenciana. Necesitan una función que clasifique automáticamente los productos según su temporada (primavera, verano, otoño, invierno) para ayudar a gestionar los pedidos.
- ▶ **Contenidos trabajados:**
  - ▶ 3.1 Estructuras de selección
  - ▶ 3.6 Codificación de programas con estructuras de control
- ▶ **Tarea:**
  - ▶ Implementa una función “clasificarTemporada” (String producto) que, mediante estructuras de selección (if, switch), imprima la temporada en la que se suele cultivar un producto.

# Actividad 2

- ▶ **Actividad 2: Cálculo de producción semanal en cooperativa de cítricos**
- ▶ **Contexto laboral:**
  - ▶ En una empresa de distribución de cítricos valencianos necesitas calcular cuántos kilos de naranjas se recogen en una semana. Cada día se registra un valor. Utilizarás una estructura repetitiva para ello.
- ▶ **Contenidos trabajados:**
  - ▶ 3.2 Estructuras de repetición
  - ▶ 3.6 Codificación de programas con estructuras de control
- ▶ **Tarea:**
  - ▶ Usa un bucle for para simular la entrada de producción diaria durante 7 días, y calcula el total de producción semanal.