

## ● Condiciones de entrega

- El plazo para la realizar las actividades de desarrollo es el que está establecido en la plataforma AULES. No se admitirán ejercicios entregados después de ese plazo.
- La entrega de todas las actividades se hará a través de la herramienta Aules y de GitHub. Debes entregar en aules el enlace y en Github en el repositorio ED un directorio con el nombre indicado a continuación.
- Cada actividad se nombrará con el nombre y primer apellido del alumno del alumno y la cadena “-actividades1-UT5”. Por ejemplo, el alumno con *Pepe Pérez*, nombrará sus actividades de la siguiente *PepePérez-actividades1-UT6*.

## ● Condiciones de corrección

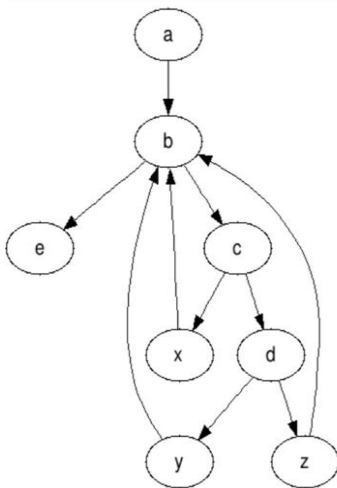
- Si se detecta copia en alguna actividad se suspenderá automáticamente la unidad de trabajo a todos los alumnos implicados.
- Las actividades se realizarán en un editor con la herramienta JavaDoc, Doxygen, Vstudio Code y/o atom.

## ● Calificación

- Las actividades puntuarán dentro del apartado de procedimientos, que supone un 15% de la nota de la UT6.

**Ejercicio 1.** Dado el siguiente pseudocódigo:

1. Obtener el grafo de flujo relacionado al anterior pseudocódigo



a = leer llaves  
b=  $x1 < x2 - 1$   
c= Lista[ix] == llave  
d= Lista[ix] < llave  
e= return  
x= resp=ix  
y=  $x1 = ix$   
z=  $x2 = ix$

2. Calcular la complejidad ciclomática

$$V(G) = 10 \text{ aristas} - 8 \text{ nodos} + 2 = 4$$
$$V(G) = \text{numero de regiones} + \text{exterior} = 3 + 1 = 4$$

Tenemos 4 caminos

**Ejercicio 2.** Realiza la prueba de la siguiente condición usando las tres técnicas conocidas:

**Por decisiones**

IF verdadero:  $x = 5, y = 30$

IF falso:  $x = 20, y = 30$

**Cobertura de condiciones**

# Pruebas

Caso 1:  $x = 5$ ,  $y = 30$  (resultado if = V) Caso 2:  $x = 30$ ,  $y = 5$  (resultado if = F)

## ***Cobertura de condiciones múltiple***

Caso 1 (Verdadero-Verdadero):  $x = 5$ ,  $y = 30$  (IF = V)  
Caso 2 (Verdadero-Falso):  $x = 5$ ,  $y = 5$  (IF = F)  
Caso 3 (Falso-Verdadero):  $x = 30$ ,  $y = 30$  (IF = F)  
Caso 4 (Falso-Falso):  $x = 30$ ,  $y = 5$  (IF = F)

**Ejercicio 3.** Realiza la prueba de bucles del siguiente fragmento. La inicialización de s es un dato de entrada que se le pregunta al usuario:

**Primero saltamos el bucle.**  
**Podemos pasar solo una vez o varias veces.**  
**Primera vez sería s=9**  
**segunda s= 10**

**Ejercicio 4.** Se va a realizar una entrada de datos de un empleado por pantalla gráfica, se definen 3 campos de entrada y una lista para elegir el oficio.

La aplicación acepta los datos de esta manera:

- Empleado: número de 3 dígitos que no empiece por 0
- Departamento: en blanco o número de dos dígitos
- Oficio: Analista, Diseñador, Programador o Elige oficio

Si la entrada es correcta el programa asigna un salario (que se muestra en pantalla) a cada empleado según estas normas:

- S1 si Oficio es Analista se asigna 2500
- S2 si el Oficio es Diseñador se asigna 1500
- S3 si el Oficio es Programador se asigna 2000

Si la entrada no es correcta el programa muestra un mensaje indicando la entrada incorrecta.

- ER1 si el Empleado no es correcto
- ER2 si el Departamento no es correcto
- ER3 si no se ha elegido Oficio

Representa las clases de equivalencia para cada condición de entrada mediante una tabla,

x = Empleado = v1: valores de 3 dígitos, n1: valores de más de 3 dígitos, n2 valores demenos de 3 dígitos, n3: valores de 3 dígitos que empiezan por 0.

y = Departamento= v2: en blanco, v3: número de 2 dígitos, n4: número de un dígito, n5: número de más de dos dígitos.

z = Oficio = v4: Analista, v5: Diseñador, v6: Programador, n6: Otra

opción

Clases para x: (V1, N1, N2,N3)

Clases para y: (V2,V3,N4,N5)

Clases para z: (V4,V5,V6,N6)

Clases para los resultados:

# Pruebas

(V7,N7,N8)

<u>V1 V3 V6</u>	<u>N1 V2 V4</u>	<u>N2 V3 V5</u>	<u>N3 V3 V6</u>	<u>V1 N4 V4</u>	<u>V1 N5 V6</u>	<u>V1 V3 N6</u>
<u>V7</u>	<u>N7</u>	<u>N7</u>	<u>N7</u>	<u>N7</u>	<u>N7</u>	<u>N8</u>
<u>230</u>	<u>1234</u>	<u>23</u>	<u>23</u>	<u>123</u>	<u>230</u>	<u>536</u>
<u>12</u>	<u>32</u>	<u>23</u>	<u>40</u>	<u>7</u>	<u>135</u>	<u>5</u>
<u>Programador</u>	<u>Analista</u>	<u>Diseñador</u>	<u>Programador</u>	<u>Analista</u>	<u>Programador</u>	<u>Camarero</u>

<u>Clases de equivalencia</u>	<u>Empleado</u>	<u>Departamento</u>	<u>Oficio</u>	<u>Resultado o esperado</u>	<u>Resultado real</u>
<u>V1 V3 V6</u>	<u>230</u>	<u>12</u>	<u>Programador</u>	<u>T</u>	<u>S3</u>
<u>N1 V2 V4</u>	<u>20</u>	<u>32</u>	<u>Analista</u>	<u>F</u>	<u>ER1</u>
<u>N2 V3 V5</u>	<u>1234</u>	<u>23</u>	<u>Diseñador</u>	<u>F</u>	<u>ER1</u>
<u>N3 V3 V6</u>	<u>23</u>	<u>40</u>	<u>Programador</u>	<u>F</u>	<u>ER1</u>
<u>V1 N4 V4</u>	<u>123</u>	<u>7</u>	<u>Analista</u>	<u>F</u>	<u>ER2</u>
<u>V1 N5 V6</u>	<u>230</u>	<u>135</u>	<u>Programador</u>	<u>F</u>	<u>ER2</u>
<u>V1 V3</u>	<u>536</u>	<u>5</u>	<u>Camarero</u>	<u>F</u>	<u>ER3</u>

## Pruebas

N6					
----	--	--	--	--	--

**Ejercicio 5.** A partir de la tabla (Definición de Clases de Equivalencia) obtenida en la actividad anterior, debes obtener los casos de prueba correspondientes. Para ello, deberás de representarlos en otra tabla donde cada fila representa un caso de prueba con:

# Pruebas

- Los códigos de las clases de equivalencia que se aplican (los que asignaste en la actividad anterior, aunque también se podría haber asignado un número a cada clase)
- Los valores asignados a las condiciones de entrada
- El resultado esperado según el enunciado del problema

Para obtener los casos de prueba deberás de tener en cuenta estas dos reglas:

- Los casos de prueba deben de cubrir la mayor cantidad de clases de equivalencia válidas (podría escribirse solo uno que cubriera todas las clases de equivalencia válidas)
- Se escribe un solo caso de prueba por cada clase de equivalencia inválido. Si se prueban múltiples clases de equivalencia no válidas en el mismo caso de prueba, pueden darse alguna de estas situaciones:
  - que algunas de estas pruebas nunca se ejecuten porque la primera enmascara a las otras o
  - que alguna de estas pruebas termina la ejecución del caso de prueba

Los casos de prueba se van añadiendo a la tabla hasta que todas las clases de equivalencia válidas y no válidas hayan sido cubiertas.

<u>V1 V3 V6</u>	<u>N1 V2 V4</u>	<u>N2 V3 V5</u>	<u>N3 V3 V6</u>	<u>V1 N4 V4</u>	<u>V1 N5 V6</u>	<u>V1 V3 N6</u>
<u>V7</u>	<u>N7</u>	<u>N7</u>	<u>N7</u>	<u>N7</u>	<u>N7</u>	<u>N8</u>
<u>230</u>	<u>1234</u>	<u>23</u>	<u>23</u>	<u>123</u>	<u>230</u>	<u>536</u>
<u>12</u>	<u>32</u>	<u>23</u>	<u>40</u>	<u>7</u>	<u>135</u>	<u>5</u>
<u>Programador</u>	<u>Analista</u>	<u>Diseñador</u>	<u>Programador</u>	<u>Analista</u>	<u>Programador</u>	<u>Camarero</u>