

Nombre: ..... DNI: .....

Convocatoria: **Septiembre 1ªPP** Semana: **1ª** Tipo de Examen: **B**

- El test debe ser contestado en la hoja de lectura óptica. Sólo una de las cuatro respuestas posibles de cada pregunta es correcta.
- El test es eliminatorio y aporta un 30% de la nota final. Son necesarias 8 respuestas correctas (6 con las prácticas aprobadas) para que se corrija el ejercicio.
- La solución del ejercicio se realizará en el reverso de esta hoja. **No se corregirán hojas auxiliares.**

### ENTREGUE ÚNICAMENTE ESTA HOJA Y LA HOJA DE LECTURA ÓPTICA sin grapar

TEST (cada respuesta correcta: 1 punto; respuesta incorrecta o en blanco: 0 puntos)

1.- La reutilización se consigue con el desarrollo de programas:

- Por refinamiento usando abstracciones
- Sólo ascendente
- Sólo descendente
- Sólo ascendente o sólo descendente

2.- Los argumentos de tipo vector abierto:

- Sólo pueden pasarse por referencia
- Pueden pasarse por valor y por referencia
- Sólo pueden pasarse por valor
- Sólo pueden utilizarse en funciones

3.- Dado el siguiente fragmento de código en Modula-2:

```
....  
CASE : BOOLEAN OF
```

- ```
....  
a) Es una selección por casos  
b) Está dentro de una unión  
c) Es una selección fija  
d) Es un caso extremo
```

4.- Si  $a := \text{TRUE}$ ,  $b := \text{FALSE}$  y  $c := \text{FALSE}$ , señale la expresión verdadera:

- $\text{NOT}(\text{NOT } a \text{ OR } c \text{ OR NOT } b)$
- $\text{NOT}(\text{NOT } c \text{ OR } b \text{ AND } a)$
- $a \text{ AND NOT}(a \text{ OR NOT } c)$
- $a \text{ AND NOT } c \text{ OR } b$

5.- Dado el siguiente subprograma:

```
PROCEDURE Oper(x: REAL; VAR y: REAL): INTEGER;  
  BEGIN  
    x := x/2.0 + y*2.0;  
    y := y + x;  
    RETURN TRUNC(y);  
  END Oper;
```

Si  $a := 5.0$  y  $b := 3.0$ , los valores de  $a$  y  $b$  después de la llamada  $\text{Oper}(a,b)$  son respectivamente:

- 8.5 y 11.5
- 5.0 y 11.0
- 5.0 y 11.5
- 8.5 y 11.0

6.- ¿Cuál de las siguientes afirmaciones es FALSA?

- Con los módulos se definen tipos abstractos
- Todo dato encapsulado es de tipo abstracto
- Todo tipo opaco es tipo abstracto
- Todo tipo abstracto es dato encapsulado

7.- Dada la siguiente declaración:

```
TYPE TipoColor = (rojo, amarillo, azul, negro);
```

Señale la expresión verdadera:

- $\text{VAL}(\text{TipoColor}(\text{ORD}(\text{rojo}))) = \text{amarillo}$
- $\text{ORD}(\text{VAL}(\text{TipoColor}, 2)) = 1$
- $\text{VAL}(\text{TipoColor}, 0) = \text{rojo}$
- $\text{ORD}(\text{azul}) = 3$

8.- Un tipo opaco en Modula-2 se detalla sólo en el módulo:

- De implementación
- De definición
- Principal de programa
- Principal y en el de definición

9.- ¿Cuánto vale la variable  $x$  después de ejecutar el siguiente código?:

```
x := 0;  
FOR i := 1 TO 10 DO  
  FOR j := 1 TO 10 DO  
    FOR k := 1 TO 10 DO  
      INC(x);  
    END;  
    DEC(x);  
  END;  
  IF ODD(x) THEN DEC(x); END;  
END;
```

- 890
- 900
- 895
- 899

10.- En Modula-2, en un registro con variantes:

- Puede omitirse el nombre del discriminante
- Puede omitirse el tipo del discriminante
- Es obligatoria la alternativa final con ELSE
- El uso de variantes no consistentes con el valor del discriminante es detectado por el lenguaje

### EJERCICIO DE PROGRAMACIÓN

Una heladería ofrece cucuruchos con bolas de 25 gramos de tres sabores distintos a escoger entre chocolate, vainilla, fresa, limón, pistacho y menta. Construya un Tipo Abstracto de Datos (TAD) que almacene la cantidad de helado disponible de cada sabor e implemente las siguientes operaciones:

- **IncrementarHelado:** añade una cantidad de helado de un determinado sabor.
- **EsPosibleCucurucho:** recibe como entrada 3 sabores e indica si hay helado suficiente para confeccionar el cucurucho.
- **CucuruchosDisponibles:** imprime en pantalla qué tipos de cucuruchos pueden confeccionarse con los helados disponibles.

**RECUERDE:** La solución del ejercicio se realizará en el reverso de esta hoja. **NO se corregirá lo que exceda de este espacio.**