

Nombre: DNI:

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

- 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.- ¿Qué podemos afirmar respecto a la siguiente declaración en Modula-2?:

VAR p : POINTER TO SET OF CHAR;

- a) No es posible definir un puntero a un conjunto
- b) No es posible definir un conjunto sobre el referencial CHAR
- c) La variable p es de tipo anónimo
- d) La variable p es de tipo opaco

2.- En Modula-2, el operador MOD:

- a) Puede dar error para valores de tipo CARDINAL
- b) Nunca da error para valores de tipo INTEGER
- c) Nunca da error para valores de tipo REAL
- d) Está definido para cualquier tipo de datos

3.- Dado el siguiente módulo en Modula-2:

```
DEFINITION MODULE Segundo;  
  PROCEDURE Previo(VAR dato : tipodato);  
END Segundo.
```

- a) Es erróneo porque se debe declarar el tipodato
- b) Es correcto porque tipodato es opaco
- c) Es erróneo porque el argumento del procedimiento está pasado por referencia
- d) Es correcto porque tipodato es un tipo anónimo

4.- Respecto al siguiente fragmento correcto en Modula-2:

```
Calculo(Dato) IN Estado
```

Siempre se puede decir que:

- a) Calculo devuelve un valor BOOLEAN
- b) Dato sólo puede ser de tipo enumerado
- c) Calculo devuelve un tipo compatible con el tipo de Estado
- d) Es una expresión condicional

5.- En Modula-2, la sentencia:

```
FROM nombreUno IMPORT nombreDos;
```

Es correcta sólo si:

- a) nombreDos es un identificador de procedimiento
- b) nombreUno es un identificador de módulo
- c) nombreDos es un identificador de módulo
- d) nombreUno es un identificador de función

6.- La doble referencia se produce cuando:

- a) Un mismo elemento se utiliza con dos nombres distintos
- b) Dos punteros apuntan al mismo tipo de elemento
- c) Se produce una redefinición de elementos
- d) Dos índices indexan el mismo elemento de una formación

7.- En Modula-2, respecto a los punteros podemos decir que:

- a) No se pueden utilizar
- b) Son elementos de programación de muy alto nivel
- c) No están tipados
- d) Su valor no es representable como un número o texto

8.- La regla sintáctica para la cual es válida el valor E5.0, siendo digito ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9, es:

- a) real ::= {digito}E{digito}
- b) real ::= digito {digito} . {digito} [E digito {digito}]
- c) real ::= {digito} [E] {digito} [.] {digito}
- d) real ::= {digito} . digito [E [+ | -] digito . {digito}]

9.- Dado el procedimiento en Modula-2:

```
PROCEDURE Referir(x:REAL; VAR y:REAL): INTEGER;  
  BEGIN  
    x := FLOAT(TRUNC(x)+2);  
    y := x/2.0 + y*2.0; y := y + x;  
    RETURN TRUNC(y);  
  END Referir;
```

Si el resultado devuelto en la llamada a Referir(A,B) ha sido 20, entonces los valores pasados han sido:

- a) A := 4.5 y B := 6.0
- b) A := 4.0 y B := 6.0
- c) A := 4.5 y B := 5.5
- d) A := 4.0 y B := 5.0

10.- ¿Cuál de las siguientes declaraciones de cabecera sería la correcta para ReadString?:

- a) PROCEDURE ReadString(VAR a : ARRAY OF CHAR);
- b) PROCEDURE ReadString(VAR a : POINTER TO ARRAY OF CHAR);
- c) PROCEDURE ReadString(VAR a : ARRAY [1..80] OF CHAR);
- d) PROCEDURE ReadString(VAR a : POINTER TO CHAR);

EJERCICIO DE PROGRAMACIÓN

Realizar un TAD para gestionar discos de datos. Cada disco dispone de un total de 64 fragmentos que pueden estar vacíos, con datos o borrados. Además, y para indicar que un fragmento de disco es consecutivo a otro, se dispone por cada fragmento de un campo apuntador al fragmento que le sigue. Si no existe un fragmento continuación, el apuntador tendrá valor 65. Las operaciones que deben realizarse son: **Formatear**, que iniciará (todo el disco vacío y los apuntadores a 65) un disco recibido como parámetro, **PedirFragmento**, que recibirá un disco y un fragmento a extender y devolverá el mismo disco modificado con el apuntador del fragmento al primer fragmento libre, y **EstaLleno**, que recibirá un disco como parámetro y devolverá cierto si no existen fragmento libres y falso si quedan huecos.

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