

Nombre: DNI:

Convocatoria: Septiembre Semana: Nacional / Unión Europea 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.
- Cada respuesta correcta: 1 punto. Respuesta incorrecta o en blanco: 0 puntos.

1.- El esquema básico de selección se corresponde con el esquema de datos:

- Tupla
- Formación
- Unión
- Registro

2.- Dadas en C± las siguientes sentencias de asignación de variables del tipo enumerado moneda:

centimo = uno;
decimo = diez;
cuartillo = veinticinco;

¿Cómo sería la declaración correcta del enumerado?

- typedef enum moneda {uno, diez, veinticinco};
- typedef enum moneda = (centimo, decimo, cuartillo);
- typedef enum moneda {1, 10, 25};
- typedef enum moneda {centimo, decimo, cuartillo};

3.- Dada la declaración de procedimiento en C±:

```
void Suma(int & x, int & y);
```

y las variables de tipo entero a y b. ¿Cuál de las siguientes llamadas al procedimiento Suma es correcta?

- Suma(a,5);
- Suma(a,a);
- Suma(a+2,b+1);
- Todas las llamadas anteriores son correctas

4.- En C±, un registro con variantes:

- Es una tupla con o sin campo discriminante
- Es una unión con campo discriminante
- Es una tupla o una unión con campo discriminante
- Siempre es una unión

5.- En una expresión condicional se pueden utilizar:

- Operadores de comparación únicamente
- Operadores lógicos únicamente
- Operadores lógicos y de comparación simultáneamente
- Operadores lógicos o de comparación pero nunca ambos simultáneamente

6.- ¿Cuál de las siguientes afirmaciones es cierta?:

- El operador != tiene prioridad ante el operador +
- El operador || tiene prioridad ante el operador /
- El operador % tiene prioridad ante el operador >=
- El operador > tiene prioridad ante el operador *

7.- Un compilador es un programa que:

- Representa programas
- Ejecuta programas
- Edita programas
- Traduce programas

8.- Conviene que la cohesión interna de un módulo y el acoplamiento entre módulos sea:

- La cohesión alta y el acoplamiento alto
- La cohesión baja y el acoplamiento bajo
- La cohesión baja y el acoplamiento alto
- La cohesión alta y el acoplamiento bajo

9.- Supongamos el fragmento de código en C±:

```
typedef int Entero;  
typedef Entero *Pt_Entero;  
....  
Pt_Entero p1, p2;  
p1 = new Entero;  
p2 = p1;  
*p1 = 7;  
*p2 = 5;
```

Después de la ejecución de este fragmento:

- *p1 vale 7 y *p2 vale 5
- *p1 vale 5 y *p2 vale 5
- *p1 vale 5 y *p2 vale 7
- *p1 vale 7 y *p2 es indeterminado

10.- Para realizar una estructura de datos no acotada es necesario utilizar:

- Enumerados
- Formaciones
- Punteros
- Tablas

EJERCICIO DE PROGRAMACIÓN

Realizar en C± un tipo abstracto de datos (TAD) para manejar polinomios de grado "n" (el mayor grado será 8):

$p(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$, con $0 \leq n \leq 8$, a_i es coeficiente real y x es entero

Con las siguientes operaciones:

Sumar: dado un polinomio p(x) y un monomio $a_i x^i$, devuelve el polinomio $p(x) + a_i x^i$

Evaluar: dado un polinomio p(x) y un valor entero b, devuelve el valor de evaluar el polinomio p(x) en b, es decir, p(b).

Imprimir: dado un polinomio p(x) lo muestra por pantalla con el formato del siguiente ejemplo:

4 + 5 * x + 10 * x2 + 15 * x3 + 20 * x4