



71901020

[iH5] Material: Ninguno

Septiembre - 2012
Original

Fundamentos de Programación

901

MÚLTIPLES GRADOS

71

Duración: 120 min.

EXAMEN: Tipo A
MixtoNacional - U.E.
1º Cuatrimestre

Hoja 1 de 1

FUNDAMENTOS DE PROGRAMACIÓN. Convocatoria: **Septiembre.** Semana: **Nacional / U.E.** 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 preguntas 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. En C±, la palabra clave **extern**:

- A. Se puede utilizar para constantes, tipos y variables en el módulo de interfaz solamente
- B. Se puede utilizar para constantes, tipos y variables en cualquier módulo
- C. Sólo se utiliza para las variables en el módulo de interfaz
- D. Se puede utilizar para tipos y variables en el módulo principal

2. Dadas las variables de tipo entero a, b, c y d, después de la ejecución de las siguientes sentencias:

```
a = 4; b = 3; c = 2; d = a+b/c*3;
```

- A. La variable d vale 7
- B. La variable d vale 9
- C. La variable d vale 4
- D. La variable d vale 1

3. En C±, la directiva **#pragma once** sirve para:

- A. Evitar el conflicto de nombres en el ámbito global
- B. Implementar abstracciones como módulos
- C. Evitar la duplicación de definiciones
- D. Implementar la recursividad

4. Si introducimos como datos por teclado: 7 4.6 el siguiente programa imprime:

```
#include <stdio.h>
typedef union intFloat{
    int x; float y;
};
int main() {
    intFloat una;
    scanf("%d", &una.x);
    scanf("%f", &una.y);
    printf("%d %.1f", una.x, una.y);
}
```

- A. 7 4.6
- B. 4.6 7
- C. [valor basura] [valor basura]
- D. [valor basura] 4.6

5. Los campos de un registro:

- A. Tienen que ser registros
- B. Pueden ser formaciones
- C. Pueden ser constantes
- D. No pueden ser registros

EJERCICIO DE PROGRAMACIÓN

Realizar en C± un TAD, con fichero de interfaz y de implementación, para manejar el sistema de una lavadora. El sistema maneja cuatro estados de funcionamiento (parada, prelavado, lavado, centrifugado) y una memoria de los 10 últimos estados de la lavadora. Los subprogramas a realizar son: **Resetear**, que pone el estado de la lavadora a parado, en cualquier caso, y lo almacena en la memoria; **CambiarEstado**, que recibe el estado actual de la lavadora y el estado al que se quiere cambiar y devuelve cierto si se ha realizado el cambio de estado y falso en caso contrario (si se realiza el cambio de estado, se almacena el nuevo estado en la memoria); **ListarEstados**, que muestra por pantalla los estados almacenados en la memoria. *NOTA: Los cambios de estado permitidos son: de parada a prelavado o lavado; de prelavado a centrifugado o lavado; de lavado a centrifugado; y de centrifugado a parada.*

6. Dado el siguiente fragmento de programa en C±:

```
void Algo(uno & dos)
```

- A. Es la llamada a un subprograma con un argumento por valor
- B. Es la cabecera de una función con un argumento por referencia
- C. Es la llamada a un subprograma con dos argumentos por valor
- D. Es la cabecera de un procedimiento con un argumento por referencia

7. Dado el siguiente fragmento de programa en C±:

```
typedef Uno Dos[Tres];
```

- A. Uno es un tipo cualquiera y Dos es una variable vector
 - B. Uno es un tipo cualquiera y Tres es una constante entera
 - C. Dos es un tipo vector y Uno es una constante entera
 - D. Dos es un tipo vector y Tres es una variable entera
8. Una función produce efectos laterales cuando:
- A. No tiene argumentos
 - B. Modifica el valor de variables externas
 - C. Devuelve un entero
 - D. No devuelve nada

9. La metodología de programación estructurada utiliza la técnica de refinamientos:

- A. Por iteración
- B. Recursivos
- C. Sucesivos
- D. Por selección

10. El siguiente fragmento de código en C±:

```
x = true;
y = false;
do {
    printf("a");
    x = y || !x;
} while(y || (!x && !y));
```

Produce la impresión del carácter a:

- A. Dos veces
- B. Un número infinito de veces
- C. Una vez
- D. Tres veces