



71901020

[\*m6E+] Material: Ninguno

Septiembre - 2014  
Reserva

Fundamentos de Programación

901

GRADO EN INGENIERÍA INFORMÁTICA

71

Duración: 120 min.  
EXAMEN: Tipo -  
MixtoNacional - U.E.  
1º Cuatrimestre

Hoja 1 de 3

FUNDAMENTOS DE PROGRAMACION	MODELO 2	CONV. EXTRAORDINARIA 13/14
MATERIAL AUTORIZADO: NINGUNO		
DURACIÓN: 2 HORAS		
INSTRUCCIONES:		
<p>1) El test debe ser contestado en la hoja de marcas. Sólo una de las cuatro respuestas posibles de cada pregunta es correcta.</p> <p>2) 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.</p> <p>3) Cada respuesta correcta 1 pto. Respuesta incorrecta, doble o en blanco: 0 ptos.</p> <p>4) Puede quedarse, si lo desea, con esta hoja de examen.</p>		

### CUESTIONES DE TEST

1. La ejecución del siguiente código:

- A.- Imprime 0 -1 una sola vez
- B.- Imprime 0 -1 un número infinito de veces
- C.- No imprime nada y entra en un bucle infinito
- D.- No imprime nada

```
i = 0;
while (i<5){
  if ((i+2)%3!=0) {
    printf("%d ", i);
  }
  i=(i+1)/(i-1);
}
```

2. La búsqueda de un elemento en un vector por dicotomía:

- A.- requiere dos bucles
- B.- sólo requiere un bucle
- C.- requiere tres bucles
- D.- no requiere ningún bucle

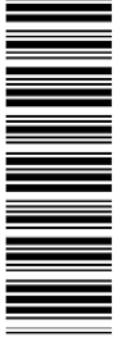
3. La ejecución del siguiente código:

- A.- Imprime 1 3 5
- B.- Imprime 1 3 5 7
- C.- Imprime 0 1 3 2 4 3 5 4 6
- D.- Imprime 1 3 2 4 3 5 4 6 5 7

```
i = 0;
while (i<7) {
  switch (%2){
    case 0: i++; break;
    case 1: i=i+2; break;
    case 2: i=i-1; break;
  }
  printf("%d ", i);
}
```

4. C± es un lenguaje de programación:

- A.- imperativo
- B.- funcional
- C.- de programación lógica
- D.- de flujo de datos



71901020

[\*m6E+] Material: Ninguno



Septiembre - 2014  
Reserva

Fundamentos de Programación

GRADO EN INGENIERÍA INFORMÁTICA

Duración: 120 min.  
EXAMEN: Tipo -  
Mixto

Nacional - U.E.  
1º Cuatrimestre

Hoja 2 de 3

901

71

5. Dada la siguiente inicialización de un vector de 6 enteros, cuál sería el programa correcto que ordena de menor a mayor el vector.

A.-

```
int main() {  
    typedef int TipoVector[6];  
    TipoVector V = {2,3,10,4,5,6};  
    int x,tmp;  
  
    for(int i=0; i < 5; i++){  
        x = i;  
        for (int j=i+1; j < 6;j++){  
            if (V[j] > V[x]){ x=j; }  
            tmp = V[i];V[i] = V[x];V[x]=tmp;  
        }  
    }  
}
```

C.-

```
int main() {  
    typedef int TipoVector[6];  
    TipoVector V = {2,3,10,4,5,6};  
    int x,tmp;  
  
    for(int i=6; i >= 1; i--){  
        x = i;  
        for (int j=5; j>=i+1 ; j--){  
            if (V[j] < V[x]){ x = j; }  
            tmp = V[i]; V[i] = V[x];V[x]=tmp;  
        }  
    }  
}
```

B.-

```
int main() {  
    typedef int TipoVector[6];  
    TipoVector V = {2,3,10,4,5,6};  
    int x,tmp;  
  
    for(int i=5; i >= 1; i--){  
        x = i;  
        for (int j=5; j>=i ; j--){  
            if (V[j] < V[x]){ x = j; }  
            tmp = V[i]; V[i] = V[x];V[x]=tmp;  
        }  
    }  
}
```

D.-

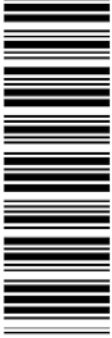

```
int main() {  
    typedef int TipoVector[6];  
    TipoVector V = {2,3,10,4,5,6};  
    int x,tmp;  
  
    for(int i=0; i < 5; i++){  
        x = i;  
        for (int j=i+1; j < 6;j++){  
            if (V[j] < V[x]){ x = j; }  
            tmp = V[i];V[i] = V[x];V[x]=tmp;  
        }  
    }  
}
```

6. Las tres unidades de compilación existentes en C++/- son:

- A.- El módulo principal, el fichero de interfaz de modulo y el de implementación del módulo
- B.- El módulo principal, el fichero de interfaz privada y la implementación del módulo
- C.- El módulo principal, el fichero de abstracción de datos y el fichero de implementación de subprogramas
- D.- No existen unidades de compilación en C++/-

7. ¿Para qué se utiliza la directiva #include?

- A.- Para salir
- B.- Para mostrar un fragmento de código
- C.- Para incluir las librerías
- D.- Para simular una entrada

 <b>71901020</b>		Fundamentos de Programación			901
		Septiembre - 2014 Reserva	GRADO EN INGENIERÍA INFORMÁTICA		71
[*m6E+] Material: Ninguno		Duración: 120 min.	EXAMEN: Tipo - Mixto	Nacional - U.E. 1º Cuatrimestre	
					Hoja 3 de 3

8. Dado el siguiente procedimiento:

```
void doble(int & a){ a=a*2;}
```

Y la llamada:

```
int b=2; doble(doble(b));
```

el valor final de b:

- A.- 8
- B.- 4
- C.- Existe un error
- D.- 16

9. En el siguiente fragmento de programa en C++:

```
#include <Uno.h>
#include "Dos.h"
```

- A.- Dos es un módulo de implementación
- B.- Uno y Dos son módulos de librería estándar
- C.- Uno es un módulo de librería estándar
- D.- Uno y Dos son módulos de implementación

10. El siguiente fragmento de programa en C++:

```
return false
```

- A.- Forma parte del bloque de un procedimiento
- B.- Forma parte del bloque de una función
- C.- Forma parte de la declaración de una función
- D.- Forma parte de la declaración de un procedimiento

## EJERCICIO DE PROGRAMACIÓN

Realizar un tipo abstracto de datos SimpleGPS (Geographical Position System) que permita manejar la información recibida de 24 satélites para determinar la posición 3D (latitud, longitud y altitud) de un objeto en el mundo. La información que se recibe de cada satélite es la distancia en metros del satélite al receptor y la calidad de la señal (buena, regular, mala). Las operaciones a resolver son: IniciarGPS, EsCalculoPosible, CalcularPosición. El procedimiento IniciarGPS inicia todo el tipo abstracto a los valores iniciales (Posición 3D a cero, y todos los satélites a distancia 0 y calidad mala). La función EsCalculoPosible devuelve cierto si podemos triangular la posición (existen datos de al menos tres satélites con calidad buena) y calcular la altura con un cuarto satélite (con datos de calidad buena). Y CalcularPosición toma los valores de tres satélites con calidad buena y llama al procedimiento Triangular para determinar la latitud y la longitud (el procedimiento Triangular pertenece a la librería cálculos.h) en un momento dado.