

Programación

Test Autoevaluación Tema 6

Autores:

M. Paz Sesmero Lorente
Paula de Toledo Heras
Fco. Javier Ordóñez Morales
Juan Gómez Romero
José A. Iglesias Martínez
José Luis Mira Peidro



Universidad
Carlos III de Madrid
www.uc3m.es



1. Dada la siguiente definición de una estructura de datos, ¿cómo se declara un vector de 10 elementos de tipo 'cancion' y se asignan valores al primer elemento?

```
struct cancion {
    char interprete[20];
    float duracion;
};
```

- No puede realizarse, un vector no puede almacenar estructuras de datos.
 - No puede realizarse, es necesario incluir valores en todos los elementos del vector.
 - ```
struct cancion descargas[10];
strcpy (descargas[0].interprete, "U2");
descargas[0].duracion = 20.7;
```
  - ```
struct cancion descargas[10];
strcpy (descargas.interprete[0], "U2");
descargas.duracion[0] = 20.7;
```
2. Dado el siguiente fragmento de código, indique cuál de las siguientes afirmaciones es cierta:

```
int main(void) {
    int lista[5]={1, 3, 5, 7, 9};
    int *p1, *p2;
    int x;
    x= *(lista);
    p1=&lista[0];
    p2=lista;
    printf("%d, %d, %d\n", x, *p1, *p2);
    return 0;
}
```

- La sentencia `x= *(lista);` genera un error de compilación porque su sintaxis es incorrecta.
 - Tras ejecutar `p1=&lista[0];` el valor almacenado en `p1` es 1.
 - La sentencia `p2=lista;` genera un error en tiempo de ejecución.
 - Tras ejecutar la sentencia `printf("%d, %d, %d\n", x, *p1, *p2);` se imprimen tres unos (1, 1, 1).
3. Indique cuál de las siguientes afirmaciones respecto a un array es incorrecta:
- Los datos de los arrays deben procesarse elemento a elemento.
 - Un array completo no puede imprimirse directamente como si fuera un bloque.
 - Un array no puede contener elementos que sean a su vez arrays.
 - Un array puede comprarse directamente con otro si los dos contienen el mismo tipo de datos y son del mismo tamaño.

4. Indique qué afirmación en relación con el siguiente programa es correcta:

```
#include <stdio.h>
struct persona{
    char nombre[50];
    int edad;
};
struct rodaje{
    char lugar[256];
    float presupuesto;
};
struct pelicula{
    struct persona director;
    struct persona actor1;
    struct persona actor2;
    struct rodaje datos;
};

int main(void)
{

    struct pelicula mi_pelicula;
    strcpy (mi_pelicula.director.nombre, "Almodovar");
    printf ("%s \n", mi_pelicula.director.nombre);
    system ("PAUSE");
    return 0;
}
```

- Se producirá un error de compilación porque la estructura persona está repetida en tres miembros de la estructura película.
- Se produce un error de compilación porque un miembro de una estructura no puede ser otra estructura.
- La sentencia `strcpy (mi_pelicula.director.nombre, "Almodovar");` genera un error en tiempo de compilación.
- Todas las afirmaciones anteriores son falsas.

5. Dado el siguiente fragmento de código de un programa principal, ¿cuál de los siguientes encabezados de función es compatible con la llamada?

```
int main(void){
    float n1,n2;
    float r1[2],r2[2];
    //.....
    n1=Calcular(r1,r2,n2);
    //.....
    return 0;
}
```

- `float Calcular (float a,float b,float c[])`
- `float Calcular (float a[],float b[],float c[])`
- `float Calcular (float a,float b,float c)`
- `float Calcular (float a[],float b[],float c)`

6. Indique la línea de código con la que se debe completar el siguiente programa en lenguaje C, para que la llamada a la función se haga correctamente.

```
#include <stdio.h>
struct fraccion {
    int num;
    int den;
};
void producto (struct fraccion f1, struct fraccion *resul,
struct fraccion f2);
int main(void)
{
    struct fraccion a={2,5}, b={3,6}, prod;
    //En este punto se añade la línea indicada
    return 0;
}
void producto (struct fraccion f1, struct fraccion *resul,
struct fraccion f2)
{
    (*resul).num = f1.num * f2.num;
    (*resul).den = f1.den * f2.den;
    return;
}
```

- `producto(a, &prod, b);`
 - `producto(a, prod, b);`
 - `producto(&a, prod, &b);`
 - Con cualquiera de las opciones anteriores se produce un error de compilación.
7. Dado el siguiente programa en C, indique cuál de las siguientes afirmaciones es correcta:

```
#include <stdio.h>

int main(void) {

    int a[10]={1,2,3,4,5,6,7,8,9,10};
    int i;

    for(i=1; i<=10; i++) {
        printf("%i ", a[i]);
    }
    return 0;
}
```

- El programa no es correcto. Se accede a una posición de memoria que no ha sido reservada para el array.
 - El programa es correcto. Imprime en pantalla todos los valores del array.
 - El programa no es correcto. La sentencia `printf("%i ", a[i]);` debería cambiarse por `printf("%i ", &a[i]);`
 - El programa no es correcto. Solo se imprimen los primeros 9 valores del array.
8. Indique cuál de las siguientes afirmaciones es cierta:
- Una estructura está compuesta por miembros que siempre son de distinto tipo.
 - Una estructura está compuesta por miembros que pueden ser de distinto tipo.
 - En la definición de una estructura cada miembro individual va entre llaves.
 - Para acceder a los miembros de una estructura se utiliza el operador (*).

9. Dado el siguiente código en lenguaje C indique qué valor tomaría la variable X.B después de ejecutar el programa:

```
#include <stdio.h>

struct ejemplo
{
    int A;
    char string [10];
    float B;
};

int main (void) {
    struct ejemplo X ={5, "string1", 3.4};
    struct ejemplo Y;
    Y.A=3;
    strcpy (Y.string, "string1");
    Y.B=5.7;
    X=Y;
    printf ("%f\n", X.B);
    system ("PAUSE");
    return 0;
}
```

- a. 3.4
 - b. 5.7
 - c. 5
 - d. 3
10. Dado el siguiente vector de enteros, indique qué instrucción imprimirá por pantalla el número 10.

```
int vector[10]={3, 20, 4, 5, -2, 6, 7, 10, 15, 0};
```

- a. printf("%d", &(vector +7));
- b. printf("%d", *(vector+7));
- c. printf("%d", &vector[7]);
- d. Ninguna de las anteriores.