

TRABAJO PRÁCTICO N° 6

Ejercicios 4

Ejercicio 1

Indica qué se escribe cuando se compila y ejecuta el siguiente programa:

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int *p, *q, *r;
    p=(int *)malloc(sizeof(int));
    q=(int *)malloc(sizeof(int));
    *q=3;
    r=q;
    *q=*q+1;
    *r=*q+1;
    *p=*q+1;
    printf("%d, %d, %d\n", *p, *q, *r);
    return 0;
}
```

- a) Escribe "5, 4, 5"
- b) Escribe "6, 6, 6"
- c) Escribe "6, 5, 5"
- d) Ninguna de las anteriores

Ejercicio 2

Indica qué se escribe cuando se compila y ejecuta el siguiente programa:

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int *p, *q, *r;
    p=(int *)malloc(sizeof(int));
    q=(int *)malloc(sizeof(int));
    *q=3;
    *r=*q;
    *q=*q+1;
    *r=*q+1;
    *p=*q+1;
    printf("%d, %d, %d\n", *p, *q, *r);
    return 0;
}
```

- a) Escribe "5, 4, 5"
- b) Escribe "6, 6, 6"
- c) Escribe "6, 5, 5"
- d) Ninguna de las anteriores

Ejercicio 3

Indica qué se escribe cuando se compila y ejecuta el siguiente programa:

```
#include <stdio.h>
int main()
{
    int x=0, *p;
    p=&x;
    printf("Introduceix un valor: ");
    scanf("%d", p);
    x=x+2;
    *p=*p+x;
    printf("%d", x);
    return 0;
}
```

- a) Si el valor introducido es 3, escribe el valor 10
- b) Si el valor introducido es 3, escribe el valor 5
- c) Si el valor introducido es 3, escribe el valor 9
- d) Se produce un error en el scanf

Ejercicio 4

Indica qué se escribe cuando se compila y ejecuta el siguiente programa:

```
#include <stdio.h>
int main()
{
    int x=0, *p;
    p=&x;
    printf("Introduceix un valor: ");
    scanf("%d", p);
    *p=*p+x;
    printf("%d", x);
    return 0;
}
```

- a) Si el valor introducido es 3, escribe el valor 0
- b) Si el valor introducido es 3, escribe el valor 6
- c) Si el valor introducido es 3, escribe el valor 3
- d) Se produce un error en el scanf

Ejercicio 5

Indica qué se escribe cuando se compila y ejecuta el siguiente programa:

```
#include <stdio.h>
int main()
{
    int a[2]={0,1};
    int *p;
    p=a;
    a[0]=2;
    a[1]=3;
    printf("%d", *p);
    return 0;
}
```

- a) 0
- b) 1
- c) 2
- d) Se produce un error en la línea p=a;

Ejercicio 6

Indica qué se escribe cuando se compila y ejecuta el siguiente programa:

```
#include <stdio.h>
void f(int a, int b)
{
    if (a>b)
        b=a;
    else
        a=b;
}
int main()
{
    int v[2]={1,2};
    f(v[1],v[0]);
    printf("%d, %d",v[0],v[1]);
    return 0;
}
```

- a) Escribe "1, 1"
- b) Escribe "1, 2"
- c) Escribe "2, 1"
- d) Escribe "2, 2"

Ejercicio 7

Indica qué realiza el siguiente programa:

```
struct punt
{
    int x;
    int y;
};
struct punt punts[10];
```

- a) Declara dos variables enteras x e y un array, el nombre del cual es punt, de 10 enteros
- b) Declara una variable, el nombre de la cual es punt, que es una estructura que tiene dos campos enteros x e y; y también un array, el nombre del cual es punts, que contiene 10 variables
- c) Declara un tipo de estructura el nombre del cual es punt, con una estructura de dos enteros x e y; y también un array, el nombre del cual es punts, que conté 10 estructuras punt
- d) Declara una estructura que contiene 10 enteros y que representa el nombre de punt

Ejercicio 8

Declarado el siguiente fragmento de código, indique cual de las siguientes afirmaciones es cierta:

```
struct punt
{
    int x;
    int y;
} p;
p punts[10];
```

- a) Declara una variable de nombre punt, que es una estructura de dos campos enteros x e y; y también una variable de tipos array, de nombre punts, que contiene 10 estructuras de tipo p
- b) Declara un tipo de estructura de nombre p, con una estructura de dos enteros x e y; i también una variable de tipo array, de nombre punts, que contiene 10 estructuras de tipo p
- c) Declara un tipo de estructura de nombre punt, con una estructura de dos enteros x e y; y también un array, el nombre del cual es punts, que contiene 10 enteros
- d) Da un error de compilación

Ejercicio 9

Del siguiente programa, indique la respuesta correcta:

```
#include <stdlib.h>
#include <stdio.h>
struct punt
{
    int x;
    int y;
};
struct punt *f()
{
    struct punt *p;
    p=(struct punt *)malloc(sizeof(struct punt));
    p->x=0;
    p->y=0;
    return p;
}
int main()
{
    struct punt *p1;
    p1=f();
    printf("%d , %d", p1.x, p1.y);
    return 0;
}
```

- a) Escribe "0, 0"
- b) Se produce un error en la línea `printf("%d , %d\n", p1.x, p1.y);`
- c) Se produce un error en la línea `p=(struct punt *)malloc(sizeof(struct punt));`
- d) Se produce un error en la línea `p1=f();`

Ejercicio 10

Del siguiente programa, indique la respuesta correcta:

```
#include <stdlib.h>
#include <stdio.h>
struct punt
{
    int x;
    int y;
};
struct punt *f()
{
    struct punt *p;
    p=(struct punt *)malloc(sizeof(struct punt));
    p->x=0;
    p->y=0;
    return p;
}
int main()
{
    struct punt *p1;
    p1=f();
    printf("%d , %d", (*p1).x, (*p1).y);
    return 0;
}
```


- a) Escribe "0, 0"
- b) Se produce un error en la línea `printf("%d ,%d" ,(*p1).x,(*p1).y);`
- c) Se produce un error en la línea `p=(struct punt *)malloc(sizeof(struct punt));`
- d) Se produce un error en la línea `p1=f();`

Ejercicio 11

Escribir una función `mayorMenor` que dados tres valores enteros, retorne un array de dos posiciones, la primera conteniendo el máximo y la segunda el mínimo de los tres.

Ejercicio 12

Escribir una función `intercambiaMayorMenor` que dado un vector de 5 enteros, intercambie la posición del máximo con la del mínimo.

Ejercicio 13

Escribir una función `mayor3a` que dado un array de tres enteros, los iguale todos al mayor.

Ejercicio 14

Escribir una función `mayor3b` que dados tres valores enteros, los iguale todos al mayor.

Ejercicio 15

Escribir una función `mayorColor` que tiene un parámetro (por referencia) del tipo color RGB. La función deberá modificar el parámetro de forma que los tres campos de la estructura contengan el máximo de entre los tres.

Ejercicio 16

Escribir un programa que lea una serie de números enteros introducidos por el usuario (de modo que el usuario teclee en primer lugar el número de enteros que va a introducir y después teclee estos enteros de uno en uno) y los almacene en una estructura de datos dinámica. Después el programa debe contar cuántos números primos contiene dicha serie.

Ejercicio 17

Escribir un programa que lea una serie de números enteros introducidos por el usuario (de modo que el usuario teclee en primer lugar el número de enteros que va a introducir y después teclee estos enteros de uno en uno) y los almacene en una estructura de datos dinámica. Después el programa debe llamar a una función donde se calculará la media de todos los valores, así como el número de valores que están por encima y por debajo de dicha media. Después de dicha llamada (desde la función `main`), el programa debe escribir los tres datos (media y número de valores por encima de la media y por debajo de la media)

Ejercicio 18

Escribir un programa que lea una serie de números enteros introducidos por el usuario (de modo que el usuario teclee en primer lugar el número de enteros que va a introducir y después teclee estos enteros de uno en uno) y los almacene en una estructura de datos dinámica. Después el programa debe dar la vuelta a la serie. Por ejemplo, si la serie original es:

0-1-2-3-4-5-6-7-8-9

acaba siendo:

9-8-7-6-5-4-3-2-1-0

Ejercicio 19

Escribir una función que permita al usuario introducir una frase (sólo con letras en minúsculas y espacios en blanco) y calcule cuántas veces aparece la palabra “hola”.

Ejercicio 20

Escribir una función que permita al usuario introducir una frase (sólo con letras en minúsculas y espacios en blanco) y calcule el número de palabras que empiezan y acaban por la misma letra.

Ejercicios

Determine cuál será el valor de las variables a y b al finalizar los siguientes programas

<p>Ejercicio 21</p> <pre>#include <stdio.h> int a,b; void f1() { b=a-1; } void f2() { b=a+b; } int main() { a=3; b=5; f1(); f2(); return 0; }</pre>	<p>Ejercicio 22</p> <pre>#include <stdio.h> int a,b; void f1(int a) { b=a-1; } void f2() { b=a+b; } int main() { a=3; b=5; f1(b); f2(); return 0; }</pre>
<p>Ejercicio 23</p> <pre>#include <stdio.h> int a,b; void f1() { b=a-1; } void f2(int b) { a=a+b; } int main() { a=3; b=5; f1(); f2(a); return 0; }</pre>	<p>Ejercicio 24</p> <pre>#include <stdio.h> void f1(int a,int b) { b=a-1; } void f2(int b, int a) { a=a+b; } int main() { int a=3,b=5; f1(b,a); f2(a,b); return 0; }</pre>
<p>Ejercicio 25</p> <pre>#include <stdio.h> void f1(int a,int *b) { *b=a-1; } void f2(int b, int *a)</pre>	<p>Ejercicio 26</p> <pre>#include <stdio.h> void f1(int *a,int *b) { *b=*a-1; } void f2(int *b, int *a)</pre>

<pre>{ *a=*a+b; } int main() { int a=3,b=5; f1(b,&a); f2(a,&b); return 0; }</pre>	<pre>{ *a=*a*b; } int main() { int a=3,b=5; f1(&b,&a); f2(&b,&a); printf("a: %d; b: %d\n",a,b); return 0; }</pre>
<p>Ejercicio 27</p> <pre>#include <stdio.h> int b; void f1(int *a) { b=*a-1; *a=b; } void f2(int *a) { *a=*a+b; } int main() { int a=3,b=5; f1(&b); f2(&a); printf("a: %d; b: %d\n",a,b); return 0; }</pre>	<p>Ejercicio 28</p> <pre>#include <stdio.h> void f1(int a, int *b) { *b=a*2; } void f2(int a[], int b) { a[0]=b; a[1]=b+1; a[2]=b+2; } int main() { int v[3]={0,1,2}; int a=3,b=5; f2(v,b); f1(v[2],&a); return 0; }</pre>
<p>Ejercicio 29</p> <pre>#include <stdio.h> void f1(int *a, int *b) { *b=*a*2; } void f2(int a[], int b) { a[0]=b; a[1]=b+1; a[2]=b+2; } int main() { int v[3]={0,1,2}; int a=3,b=5; f2(v,b); f1(&v[1],&b); return 0; }</pre>	<p>Ejercicio 30</p> <pre>#include <stdio.h> void f1(int *a, int *b) { *b=*a*2; } void f2(int a[], int *b) { a[0]=*b; a[1]=*b+1; a[2]=*b+2; *b=*b*2; } int main() { int v[3]={0,1,2}; int a=3,b=5; f2(v,&a); f1(&v[1],&b); return 0; }</pre>

Ejercicio 31

Escribir un programa que permita introducir una frase al usuario (sólo con letras en minúsculas y espacios en blanco) y que muestre por pantalla cuántas veces aparece cada letra y cuál es la que más veces aparece (en caso de empate, la primera alfabéticamente).

Ejercicio 32

Escribir un programa que permita introducir una frase al usuario (sólo con letras en minúsculas y espacios en blanco) y que muestre por pantalla cuántas palabras diferentes hay en la frase, cuál es la palabra que más veces aparece y cuántas veces lo hace.

Notas:

- En caso de empate, se seleccionará la que primero aparece en la frase
- Debés obligatoriamente definir un tipo de datos mediante estructuras para representar una palabra y el número de veces que ésta aparece. Nota que para controlar las palabras que van apareciendo deberás ir guardando esas estructuras en un array.
- Podés asumir que como máximo habrá 100 palabras diferentes en la frase, e igualmente que una palabra tendrá un máximo de 20 letras.

Ejercicio 33

Escribir un programa que permita introducir una frase al usuario (sólo con letras en minúsculas y espacios en blanco) y que muestre por pantalla qué palabras diferentes aparecen, y cuál es la letra que aparece más veces en cada una de ellas (en caso de empate, la primera alfabéticamente).

Ejercicio 34

Supongamos que tenemos cuatro observatorios meteorológicos (Capital Federal, Córdoba Capital, Neuquen, Resistencia) . Para cada uno de ellos se han recogido una serie de datos cada día del año. Dichos datos son:

- temperatura máxima (número real)
- temperatura mínima (número real)
- precipitaciones totales (número entero)
- situación general (por ejemplo “soleado”, “soleado con intervalos nubosos por la tarde”, ...)

Necesitamos un programa que permita:

1. introducir esos valores de forma manual en nuestro programa: el usuario elige el día, mes y provincia y a continuación introducirá los valores.
2. Obtener la media de las temperaturas máximas en cada ciudad para cada mes
3. Ídem para la media de las temperaturas mínimas
4. Obtener el total de precipitaciones en cada ciudad para cada mes
5. Obtener el día, mes y ciudad en que se produce la temperatura más alta del año
6. Ídem con la más baja.

Nota: para los puntos 2 a 6 hay que tener en cuenta que puede que no se hayan introducido los datos de todos los días, así que sólo deben tomarse en cuenta los valores de los días introducidos.

Webgrafía y Licencia

- ◆ Textos tomados, corregidos y modificados de diferentes páginas de Internet, tutoriales y documentos.
- ◆ Este documento se encuentra bajo Licencia Creative Commons Attribution – NonCommercial - ShareAlike 4.0 International (CC BY-NC-SA 4.0), por la cual se permite su exhibición, distribución, copia y posibilita hacer obras derivadas a partir de la misma, siempre y cuando se cite la autoría del **Prof. Matías García** y sólo podrá distribuir la obra derivada resultante bajo una licencia idéntica a ésta.

◆ Autor: **Matías E. García**

Prof. & Tec. en Informática Aplicada
www.profmatiasgarcia.com.ar
info@profmatiasgarcia.com.ar

