



TEMA 5: Subprogramas. Programación modular

EJERCICIOS DE TRAZAS

1.- Escribir la traza y decir que imprime por pantalla cada uno de los siguientes programas:

```

a.-
/*****/
/* Ejercicio de traza 1 */
/*****/

```

```

int a, b, c;

void E1();

int main()
{
1   a = 11;
2   b = 12;
3   c = 13;
4   E1();
    cout << a << b << c;

    return 0;
}

void E1()
{
    int b;

5   a = 1;
6   b = 2;
7   c = 3;

    return;
}

```

	<i>a</i>	<i>b</i>	<i>c</i>	
1	11	?	?	
2	11	12	?	
3	11	12	13	<i>b</i> ₁
4	11	12	13	?
5 ₁	1	12	13	?
6 ₁	1	12	13	2
7 ₁	1	12	3	2
4	1	12	3	



b.-

```

/*****
/* Ejercicio de traza 2          */
/*****

int a, b;

void E2(int a, int b);

int main()
{
1   a = 1;
2   b = 2;
3   E2(b, a);
    cout << a << b;

    return 0;
}

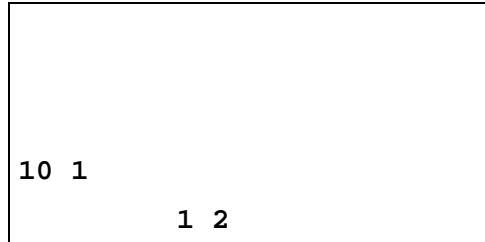
void E2(int a, int b)
{
4   a = 10;
    cout << a << b;

    return;
}

```

	<i>a</i>	<i>b</i>		
<i>1</i>	1	?		
<i>2</i>	1	2	<i>a₁</i>	<i>b₁</i>
<i>3₁</i>	1	2	2	1
<i>4₁</i>	11	12	10	1
<i>3</i>	1	2		

Pantalla





C.-

```

/*****/
/* Ejercicio de traza 3 */
/*****/

int y, z;

void Cambiar(int & i, int j);

int main()
{
1   y = 21;
2   z = 7;
3   Cambiar(y, z);
4   Cambiar(z, y);
    cout << 'z=' << z << ' y=' << y;

    return 0;
}

void Cambiar(int & i, int j)
{
    int k;

5   k = j;
6   i = k + j;
7   k = i;

    return;
}

```

	y	z			
1	21	?			
2	21	7	i_1/y	j_1	k_1
3	21	7	21	7	?
5 ₁	21	7	21	7	7
6 ₁	14	7	14	7	7
7 ₁	14	7	14	7	14
3	14	7	i_2/z	j_2	k_2
4	14	7	7	14	?
5 ₂	14	7	7	14	14
6 ₂	14	28	28	14	14
7 ₂	14	28	28	14	28
4	14	28			

Pantalla

z=28 y=14



d.-

```

/*****/
/* Ejercicio de traza 4 */
/*****/

int x;

void Calculo();

int main()
{
1   x = 10;
2   Calculo();
   cout << x;

   return 0;
}

void Calculo()
{
3   x = x + 7;

   return;
}

```

	x	
1	10	x_1
2	10	?
3 ₁	10	?+7
2	10	

Pantalla

10



e.-

```

/*****/
/* Ejercicio de traza 5 */
/*****/

int x;

void Calculo(int & n);

int main()
{
1   x = 20;
2   Calculo(x);
   cout << x;

   return 0;
}

void Calculo(int & n)
{
3   x = x + 5;
4   n = n + 3;

   return;
}

```

	<i>x</i>	
1	20	n_1/x
2	20	20
3 ₁	25	25
4 ₁	28	28
2	28	

Pantalla

28



2.- Dados los siguientes programas, realiza sus trazas y di que muestran por pantalla:

a.-

```
#include <iostream.h>

int Funcion (int y, int & x)
{
    int z;

    z = x + 1;
    x += 3;
    y = (z - y) * 2;

    return y;
}

int main(void)
{
    int x, y, z;

    x = 1;
    y = -2;

    z = Funcion (x, y);
    y = Funcion (x, z);

    cout << x << y << z;

    return 0;
}
```

b.-

```
#include <iostream.h>

int Funcion (int & y, int x)
{
    int aux;

    x += 2;
    y = (x - y) * 3;

    return y;
}

int main(void)
{
    int x, y, z;

    x = 1;
    y = -2;

    z = Funcion (x, y);
    y = Funcion (x, z);

    cout << x << y << z;

    return 0;
}
```

3.- Dados los siguientes programas, realiza sus trazas:

a.-

```
#include <iostream.h>

int Funcion (int y)
{
    int x, aux;

    if (y < 0)
        x = 1;
    else
    {
        aux = Funcion (y - 2);
        x = aux + 2 * (y - 1) + 1;
    }

    return x;
}

int main(void)
{
    int x, y;

    x = 5;

    y = Funcion (x);

    return 0;
}
```

b.-

```
#include <iostream.h>

int Funcion (int y)
{
    int x, aux;

    if (y == 0)
        x = 0;
    else
    {
        aux = Funcion (y - 1);
        x = aux + 2 * (y - 1) + 1;
    }

    return x;
}

int main(void)
{
    int x, y;

    x = 3;

    y = Funcion (x);

    return 0;
}
```

**4.- Realiza la traza del siguiente programa:**

```
#include <iostream.h>
#include <stdlib.h>

int opera (int a, int b, int & d)
{
    int c = 0;

    if (a < b)
        c = opera (b, a, d);
    else
    {
        while (a > b)
        {
            a -= b;
            c++;
        }
        d = a;
    }
    return c;
}

int main()
{
    int a = 5;
    int b = 16;
    int c, d;

    d = opera (a, b, c);

    cout << "a y b: " << a << ", " << b << endl;
    cout << "c y d: " << c << ", " << d << endl;

    return 0;
}
```

5.- Realiza la traza del siguiente programa y averigua qué se escribe en pantalla.

```
float a;

void f1(int &x, int &y)
{
    int z = x;
    x = (y + x) / 2 + a;
    y = z - y;
    a = a - y;
    return;
}

int f2 (int &y, int x)
{
    int a;
    a = x;
    x = x - y / 2;
    y = a + y / 2;
    a = x;
    x = x + y;
    return a;
}

void main()
{
    int y, z;

    z = 3;
    y = 5;
    a = 10;

    f1 (y, z);
    a = a + z;
    y = f2 (z, y);
    cout << y - a << " : " << z - a << endl;
    return;
}
```


**6.- Realiza las trazas de los siguientes programas y di que muestran por pantalla:**

```
#include <iostream.h>
#include <stdlib.h>

int f(int);
int main(void);

int main(void)
{
    int a, b;

    b = 3;
    a = f(b);

    cout << a;
}

int f(int a)
{
    int b, c, d;

    if (a == 0)
        d = 0;
    else
        if (a == 1)
            d = 1;
        else
            {
                b = f(a - 1);
                c = f(a - 2);
                d = b + c;
            }
    return d;
}
```

```
#include <iostream.h>
#include <stdlib.h>

int const BASE = 5;

int main (void);
int Examen (int);

int main(void)
{
    int x, y;

    x = 10;

    y = Examen (x);

    cout << "y = " << y << endl;

    system("PAUSE");
    return 0;
}

int Examen (int y)
{
    int x;

    if (y == 0)
        x = 0;
    else
        {
            x = Examen (y / BASE);
            x = x * 10 + y % BASE;
        }

    return x;
}
```

7.- Realiza la traza del siguiente programa y di que muestra por pantalla:

```
#include <iostream.h>

int x;

int f(int x, int & y);

int f(int x, int & y)
{
    y = y + x;
    x = x - 1;
    return x;
}
```

```
int main(void)
{
    int a;

    x = 2;

    a = f(x - 1,x);

    cout << a << x;
}
```



8.- Realiza la traza del siguiente programa y di que muestra por pantalla:

```
#include <iostream.h>

int f(int a, int b);

int main(void)
{
    int a;

    a = f(1,1);

    cout << a;
}
```

```
int f(int a, int b)
{
    int r;

    if(a + b == 0)
        r = 0;
    else
    {
        r = f(2 * a, b - a - 1);
        r = r + a;
    }
    return r;
}
```

9.- Dado el siguiente programa:

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

int f(int x, int & y)
{
    x = x + 4;
    y = y + 2;
}
```

```
int main(void)
{
    int x = 1; int y = 2;

    f(y, x);
    y = y + x;
    cout << "x:" << x << "y:" << y;

    return 0;
}
```

¿Qué valores visualizará para x e y?

10.- Dado el siguiente programa:

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

int main(void)
{
    int x = 1; int y = 2;

    f(y, x);
    y = y + x;
    cout << y;
    return 0;
}

void f(int & x, int y)
{
    x = x + 3;
    y = y + 1;
}
```

¿Qué visualizará? Elegir una opción:

a)	3	b)	6	c)	7	d)	8
----	---	----	---	----	---	----	---

**10.-** Dados los siguientes programas:

```
#include<iostream.h>
#include<string>

int main (void);
string f (string s);

string f(string s)
{
    char c;
    string res;

    if(s == "")
        res = "";
    else
    {
        c = s[0];
        s.erase(0,1);
        res = f(s);
        res = res + c;
    }
    return res;
}

int main(void)
{
    string s = "hola";

    cout << f(s);

    return 0;
}
```

```
#include<iostream.h>
#include<string>

string f(char c, string s);
int main(void);

string f(char c, string s)
{
    string s2, s3;
    if (s!="")
    {
        s2 = s.substr(1, s.length()-1);
        s3 = f(c,s2);
        if (s[0] == c)
            s = s3;
        else
            s = s[0] + s3;
    }
    return s;
}

int main(void)
{
    string s;

    s = "ana";
    s = f('a', s);
    cout << s;

    return 0;
}
```

Realiza su traza y di que mostrarán por pantalla.

erase(pos, num): Método que elimina **num** caracteres de un **string** empezando en la posición **pos** (inclusive).

length(): Método que devuelve el número de caracteres del *string*.

substr(ini, num): Método que devuelve un substring que empieza en la posición **ini** (inclusive) y contiene **num** caracteres.

11.- Sea el siguiente programa en C++:

```
#include<iostream.h>

int main (void);
int f (int n);

int main(void)
{
    cout << f(1101);
    return 0;
}
```

```
int f(int n)
{
    int a;
    int res;

    if(n <= 1)
        res = n;
    else
    {
        a = f(n / 10) * 2;
        res = a + n % 10;
    }
    return res;
}
```

Realiza una traza del programa y di que mostrará por pantalla.



12.- Sea el siguiente programa en C++:

```
int f(int & x);

int main(void)
{
    int x = 1; int y = 2;

    y = y + f(x);
    y = y + x;
    cout << y;
    return 0;
}

int f(int & x)
{
    x = x + 2;
    return(x - 2);
}
```

Escribe qué muestra por pantalla.

13.- Sea el siguiente programa en C++:

```
int f(int a)
{
    int b,c;

    if (a <= 1)
        c = a;
    else
    {
        b = a % 2;
        a = a / 2;
        c = f(a);
        c = c * 10 + b;
    }
    return c;
}
```

```
int main(void)
{
    int x;

    x = f(5);
    cout << x << endl;

    return 0;
}
```

Realiza una traza del programa y di cuál será el valor final de x.