

CUESTIONES PROPUESTAS

1. Escriba la traza y diga qué imprime por pantalla cada uno de los siguientes programas:

```

/*****/
/* Ejercicio de traza */
/*****/
public class Traza {
    public int F(int a, int b)
    {
        int resto;
        int resultado;
        resto = a % b;
        if (resto == 0)
            resultado = b;
        else
            resultado = F(b, resto);
        return resultado;
    }
    public static void main (String[] args){
    {
        int z;
        z = F(12, 15);
        System.out.println(z);
    }
}

```

2. Indique cuál es el atributo de instancia y cuál es el atributo de clase del siguiente código:

```

public class unaClase {
    public static int a = 20;
    public int b = 12;
}

```

3. ¿Qué mostrará el siguiente programa por pantalla?

```

public class Bebe {

    Bebe(int i) {
        this("Soy un bebe consentido");
        System.out.println("Hola, tengo " + i + " meses");
    }
    Bebe(String s) {
        System.out.println(s);
    }
    void berrea() {
        System.out.println("Buaaaaaaaaaaaaa");
    }
    public static void main (String[] args){
    {
        New bebe(8).berrea;
    }
}

```

PROGRAMAS PROPUESTOS

1. Cree un paquete Utilidades.mates con dos clases sumar y potenciar. La clase sumar tendrá un método `int suma(int,int)` el cual devolverá la suma de dos parámetros introducidos y la clase potenciar tendrá un método `int potencia(int, int)` el cual devolverá el resultado de elevar el primer parámetro al segundo parámetro. Realice un programa que haga uso de este paquete.
2. Realice la clase Pez, la cual tendrá un miembro nombre de tipo String el cual podrá ser heredado por sus subclases. Realice un método `getNombre` y otro `setNombre`. Utilice el objeto `this` en estos métodos. Implemente en esta clase el método `clone()` así como el método `equals()` para poder hacer una

comparación en profundidad. Realice un programa que haga un testeo en profundidad de las características de la clase.

3. Para la clase anterior Pez, cree un miembro privado entero numpeces común a todos los objetos pez el cual cuente el número de peces creados. Crea un programa que compruebe que esta variable se incrementa cada vez que se crea un objeto pez.
4. Para el objeto pez anterior cree un constructor copia. Compruebe este constructor mediante un programa.
5. Cree una clase prueba que tenga dos métodos (primero y segundo). El método segundo llamará al método primero dos veces, de forma normal y utilizando this. Verifica que ambas llamadas son equivalentes.
6. El número de combinaciones de m elementos tomados de n en n es:

$$\binom{m}{n} = \frac{m!}{n!(m-n)!} \quad \text{donde } m! = m * (m-1) * \dots * 1$$

Realice un programa que lea desde el teclado los valores de m y n. comprobar que m es mayor que n y calcular el número de combinaciones.

7. Escriba un programa que lea varias líneas de texto y determine la media de caracteres (incluyendo signos de puntuación y espacios en blanco) en cada línea. Estructure el programa de modo que se continúen leyendo líneas hasta encontrar una línea en blanco (es decir, el primer carácter es \n).
8. Diseñe un programa que dada una cantidad a pagar y el dinero con el que se paga, devuelva desglosado en billetes y monedas el dinero que sobre.

PROGRAMAS RECURSIVOS

9. Implemente un subprograma que realice la serie de Fibonacci, que es:

Fibonacci (1) = Fibonacci (2) = 1

$N > 2$ Fibonacci (n) = Fibonacci (n-1) + Fibonacci (n-2)

10. Realice mediante funciones recursivas un programa que traslade un número entero en base diez a cualquier otra base.
11. Escriba una función llamada potencia que tenga de argumentos de entrada x y n, devuelva x elevado a n. El argumento x es de tipo double y el argumento n es de tipo int.
12. El máximo común divisor (m.c.d.) entre dos números enteros mayores que cero viene dado por la siguiente forma:

$$m.c.d.(m, n) = \begin{cases} n, \text{ si } m \% n = 0 \\ m.c.d.(n, r), \text{ si } m \% n = r, r > 0 \end{cases}$$

Escriba un programa que calcule el máximo común divisor entre dos números.

13. Para obtener el número de tarot de una persona, hay que sumar los números de su fecha de nacimiento y reducirlos a un solo dígito. Realice un programa que lea una fecha de teclado y escriba el número del tarot a partir de la fecha leída. La fecha estará formada por tres números enteros, el día, el mes y el año (4 dígitos). Ejemplo: Supóngase que una persona nace el día 1 de julio de 1966. La suma $1+7+1996=1974$. El resultado obtenido no está formado por un solo dígito, por lo que habrá que sumar las cuatro cifras que componen el número: $1+9+7+4=21$. Al igual que antes, el resultado no está formado por un dígito por lo que repetir el proceso, $2+1=3$, El resultado obtenido es el número del tarot 3.
14. Implemente un subprograma que halle cual es la primera potencia en base 2 mayor que un número que pasamos como parámetro, devolviendo el valor de dicha potencia y el exponente al que está elevado.
15. Implemente un subprograma que calcule recursivamente en cuanto se convierte un capital C al final de N años y a un interés I.
16. Calcule el primer término de la siguiente serie que sea mayor o igual a un valor V que se le pasa como parámetro y devuelva el lugar que ocupa en la serie y el valor.
 $A_1 = 0$
 $A_n = n + (A_{n-1})!$
17. Calcule el valor de la serie donde n es un valor que se pasa como parámetro al subprograma

que hace el cálculo.
$$\sum_{i=0}^{n-1} (1 + i * \sum_{j=0}^n \frac{1}{n})$$

1. Sea el siguiente programa:

```
void Calculo()
{
    int x;
    x = x + 7;
    return;
}
int main()
{
    int x;
    x = 10;
    Calculo();
    cout << x;
    return 0;
}
```

¿Cuál es su salida por pantalla? _____

2. Considere la siguiente variación del programa anterior:

```
void Calculo(int & n)
{
    x = x + 5;
    n = n + 3;
    return;
}
int main()
{
    int x;
    x = 20;
    Calculo(x);
    cout << x;
    return 0;
}
```

¿Cuál será ahora su salida? _____

3. Sea el siguiente programa:

```
int f(int n);
{
    n = n + 4;
    return n;
}
int main()
{
    int x;
    x = 6;
    cout << f(x) << ' ' << x;
}
}
```

¿Cuál de las siguientes salidas por pantalla corresponde con la ejecución del programa?

- (a) 10 10
- (b) 10 6
- (c) 6 10
- (d) 6 6

4. Dado el siguiente programa:

```
void que_cosa(int & x, int & y)
{
    x = y + 2;
    y = x + 2;
}
int main()
{
}
```

```

        int x, y;
        x = 1; y = 2;
        que_cosa(y, x);
        cout << x << " " << y;
    }

```

¿Qué se visualiza al ejecutar este programa? _____

5. Haz la traza del siguiente programa y averigua que se visualiza al ejecutarse:

```

int func7(int b, int e)
{
    int res;
    if (e==1)
        res = b;
    else
        res = b * func7(b, e-1);
    return res;
}

int main()
{
    int r;
    r = func7(2, 3);
    cout << r;
    return 0;
}

```

6. Haz la traza del siguiente programa y averigua que se visualiza al ejecutarse:

```

int Funcion5 (int x, int y)
{
    int z;
    if (x > y)
        z = Funcion5 (y, x - y) + y;
    else
    {
        if ( y < 0 )
            z = Funcion5 ( -y, x ) + x;
        else
            z = 0;
        return z;
    }
}

int main (void)
{
    int a;
    a = Funcion5 (-2, -1);
    cout << a << endl;
    system("pause");
    return 0;
}

```