

Programación

Tema 4



Indice

Portada (Pagina 1)

Ejercicios (Pagina 3 – 6)

Conclusión (Pagina 7)

1. Crear una función llamada mayor que indique si un número entero es mayor que otro.

2. Realizar 2 llamadas a la función anterior con los valores 4,5 y 6,2

```
int mayor (int n1,int n2){
    return n1>n2;
}
int main()
{
    cout << "El resultado de mayor es "<< mayor(4,5)<<endl;
    cout << "El resultado de mayor es "<< mayor(6,2)<<endl;

}
```

3. Crear una función llamada mayor que devuelve el mayor de 3 caracteres en el mismo fichero que la función del ejercicio 1.

4. Usar la función del ejercicio 3 con los números 5,6,1 y 4,88,64

```
int temporal=0;
int mayor (int n1,int n2,int n3){
    if (n1>n2 && n1 > n3){
        temporal=n1;
    }else if (n2>n1 && n2>n3){
        temporal=n2;
    }else{
        temporal=n3;
    }
    return temporal;
}
int main()
{
    cout << "El resultado de mayor es "<< mayor(5,6,1)<<endl;
    cout << "El resultado de mayor es "<< mayor(4,88,64)<<endl;

}
```

5. Crear una función que descomponga un número en factores primos, es necesario además crear otra función que indique si un número es primo o no, por ejemplo al descomponer 45, se empieza por el 2 que es primo (llamara a esprimo(2)) pero 45 no es divisible por 2, pasando al 3, que es primo (se comprueba) y 45 es divisible por 3, quedando 15 y mostrando el 3, a continuación se pasa a 4 y se comprueba si es primo (no lo es), con lo que se pasa al 5....

```
int temporal=0;
int primo (int n1){
    int i=2;
    int primo=0;
    for (i=2;i<n1 && primo==0;i++){
        if(n1%i==0){
            cout<< "el numero no es primo "<< n1<<endl ;
            primo=1;

        }else{
            cout<< "el numero es primo "<< n1<<endl ;
            primo=0;

        }
    }
    return n1;
}
int main()
{
    cout << " "<< primo(13)<<endl;

}
```

Esta a medias

7. Crear una función que devuelve el área de un círculo con valores reales y probarlo

```
#include <iostream>

using namespace std;
const float PI = 3.1416;
float area(int radio){
float temporal=0;
temporal=PI*radio*radio;
return temporal;
}
int main()
{
cout << "El resultado del area es " << area(5) << endl;

}
```

8. Implementar una función que imprima un vector de enteros, recordar que no se tiene el atributo length en C

```
int mostrar_vector(int* vector, int tamanio) {
int i=0;
int temporal=0;
for (i = 0; i < tamanio; i++) {
cout << "" << vector[i] << "\n";
}
}

int main()
{
int tamanio=5;
int vector[tamanio] = {1,2,3,4,5};
cout << " " << mostrar_vector(vector,tamanio) << endl ;
}
```

10. Sin escribir el código de la siguiente función indicar que valen las variables pasadas como parámetros al salir de la misma.

a=4

b=7

Al ser por valor cuando sale de la función el valor de a y b no cambia.

11. Realizar el mismo análisis que en el ejercicio anterior.

a=5

b=6

Al pasarlo por referencia el valor si cambia

12. . Analizar las diferencias entre los dos ejercicios anteriores.

Uno pasa por valor y otro por referencia por lo que en uno cuando la función acaba no modifica los valores fuera y en otro al modificar la dirección de memoria si lo hace

13. Crear una función recursiva que calcule eleve un número a otro y probar. Definir en primer lugar caso base y caso recursivo en papel

```
int elevar(int base, int exponente) {  
    if (exponente == 0) {  
        return 1;  
    }  
    return base * elevar(base, exponente - 1);  
}  
int main() {  
    cout << elevar(5, 2) << endl;  
  
}
```

14. Definir una función que indique si un número es primo o no, de forma recursiva. Definir en primer lugar caso base y caso recursivo en papel.

Conclusión	
Dificultad	★ ★ ★ ★ ★
Interesante	★ ★ ★ ★ ★
Rápido	★ ★ ★ ★ ★

