# Funciones

Objetivo: conocer el uso de las funciones

Tiempo de realización: 1 semana

Para cada uno de los ejercicios indicados abajo, cree tanto la función, como su función main para probarla.

1 ) **Nombre**: espar

**Parámetros**:

v: entero: valor para calcular si es o no par

**Retorno**: boleano: valor boleano indicando si el numero es par( true) o no (false)

**Explicación**: La función recibe como parámetro un valor v, y retorna true si es par, y false en caso contrario.

Solución:

#include <iostream>

using namespace std;

bool espar(int v){

bool res;

if (v%2==0){res=true;}

else {res=false;}

return res;

}

int main(){

int n;

cout<<"Introduzca un valor"<<endl;

cin>>n;

bool ep=espar(n);

if ( ep==true ){

cout<<"El valor es par"<<endl;

}

else{

cout<<"El valor no es par"<<endl;

}

}

2) **Nombre**: esprimo

**Parámetros**:

v: entero: valor para calcular si es o no primo

**Retorno**: boleano: Retorna true si el número es primo, y false en caso contrario.

**Explicación**: La función indica si el número pasado es o no primo.

3) **Nombre**: factorial

**Explicación**: La función retorna el factorial del valor *v* pasado como parámetro

**Parámetros**:

v: entero: valor sobre el que calcular el factorial

**Retorno**: entero: factorial de v

4) **Nombre**: divisores

**Parámetros**:

v: entero: valor sobre el que calcular los divisores

**Retorno**: nada (void)

**Explicación**: La función imprime por pantalla los divisores de un número v en orden ascendente.

5) **Nombre**: primos

**Parámetros**:

n: entero: número de primos

**Retorno**: nada (void)

**Explicación**: La función imprime por pantalla los n primeros números primos. La función deberá llamar a la función esprimo del ejercicio 2.

Por ejemplo si n=5, los primeros 5 primeros numeros primos son el 2,3,5,7 y el 11

6) **Nombre**: **combinaciones**

**Explicación**: Calcula las combinación de n sobre k. Lease <https://es.wikipedia.org/wiki/Coeficiente_binomial> para más información

**Parámetros**:

n: entero

k: entero

**Retorno**: numero combinatorial de n sobre k

7) **Nombre**: Ndivisores

**Explicación**: La función devuelve el número de divisores del número v pasado como parámetro.

**Parámetros**:

v: entero: valor sobre el que calcular los divisores

**Retorno**: entero

8) **Nombre**: esperfecto

**Explicación**: La función calcula si el número pasado como parámetro es perfecto o no.

**Parámetros**:

v: entero: valor sobre el que calcular si es perfecto

**Retorno**: bool . true si el número es perfecto y false en caso contrario