

```
int i  
#include<iostream>
```

```
Return 0
```

```
While(i <= j) {  
    }
```

```
if(x > 4) {  
    }
```

```
10100100100
```

```
00001100100
```

```
11101010010
```

```
0101
```

```
0  
1  
1  
1  
1  
0
```

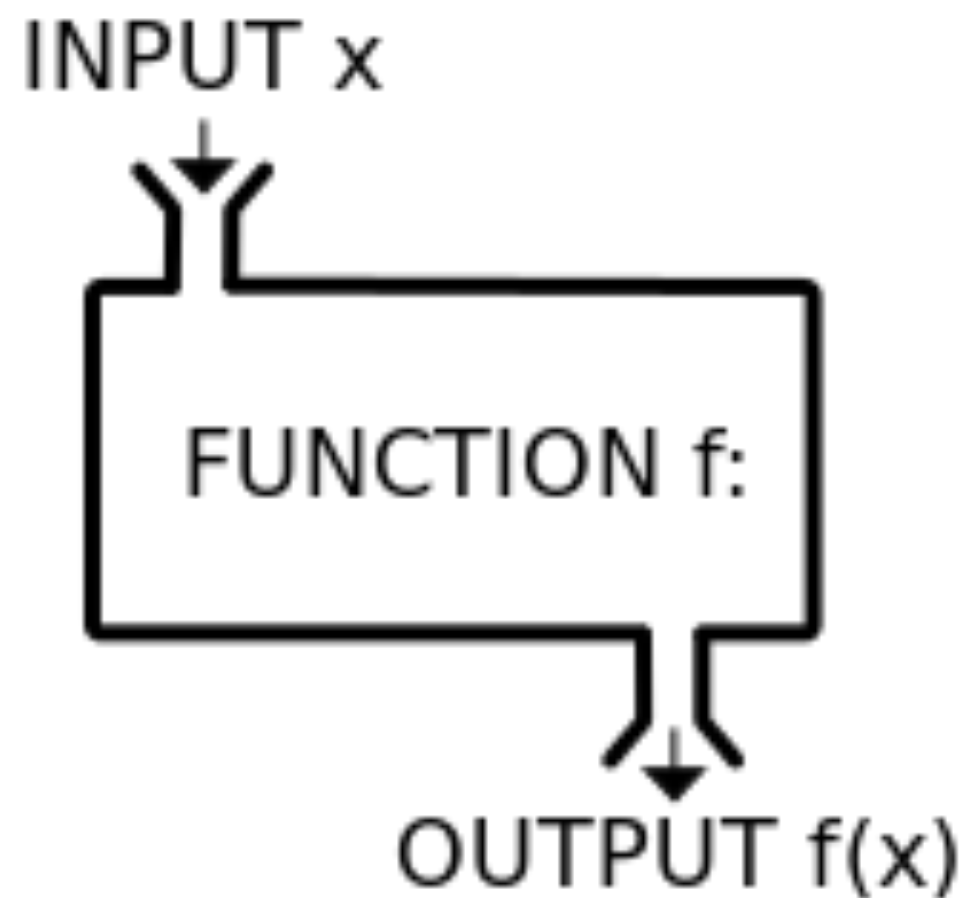
Funciones

Prof. Nicolás Hidalgo
nicolas.hidalgoc@mail.udp.cl



Funciones

- ¿Qué es una función en programación?



Funciones

- En programación, una función es un bloque de código que tiene una finalidad específica, independiente del resto del programa
- Posee 4 componentes principales:
 - **Nombre:** nombre de la función con el que se identifica y se distingue de otras.
 - **Parámetros:** que son los valores que recibe la función como entrada
 - **Código de la función:** que son las operaciones que hace la función
 - **Resultado** (o valor de retorno): que es el valor final que entrega la función

Funciones

- En esencia una función es un “mini-programa”, también se les llama **sub-rutinas** o sub-programa por lo mismo
- Posee componentes análogos a un programa: entradas, proceso, y salida
- Ojo en C++, parámetros y retorno son opcionales

Funciones

- Las funciones son muy importantes para los programas:
 - facilitan entendimiento programa
 - facilita la detección de errores
 - facilita dar solución a problemas complejos (divide y vencerás)

Funciones

Codigos — vim test.cpp — 80×46

```
#include<iostream>
using namespace std;

int area (int a){
    return a*a;
}

int main (){
    int lado;

    cout << "Ingresa el lado del cubo: ";
    cin >> lado;

    cout << "Area Cubo de lado " << lado << " es: " << area(lado)<<endl;

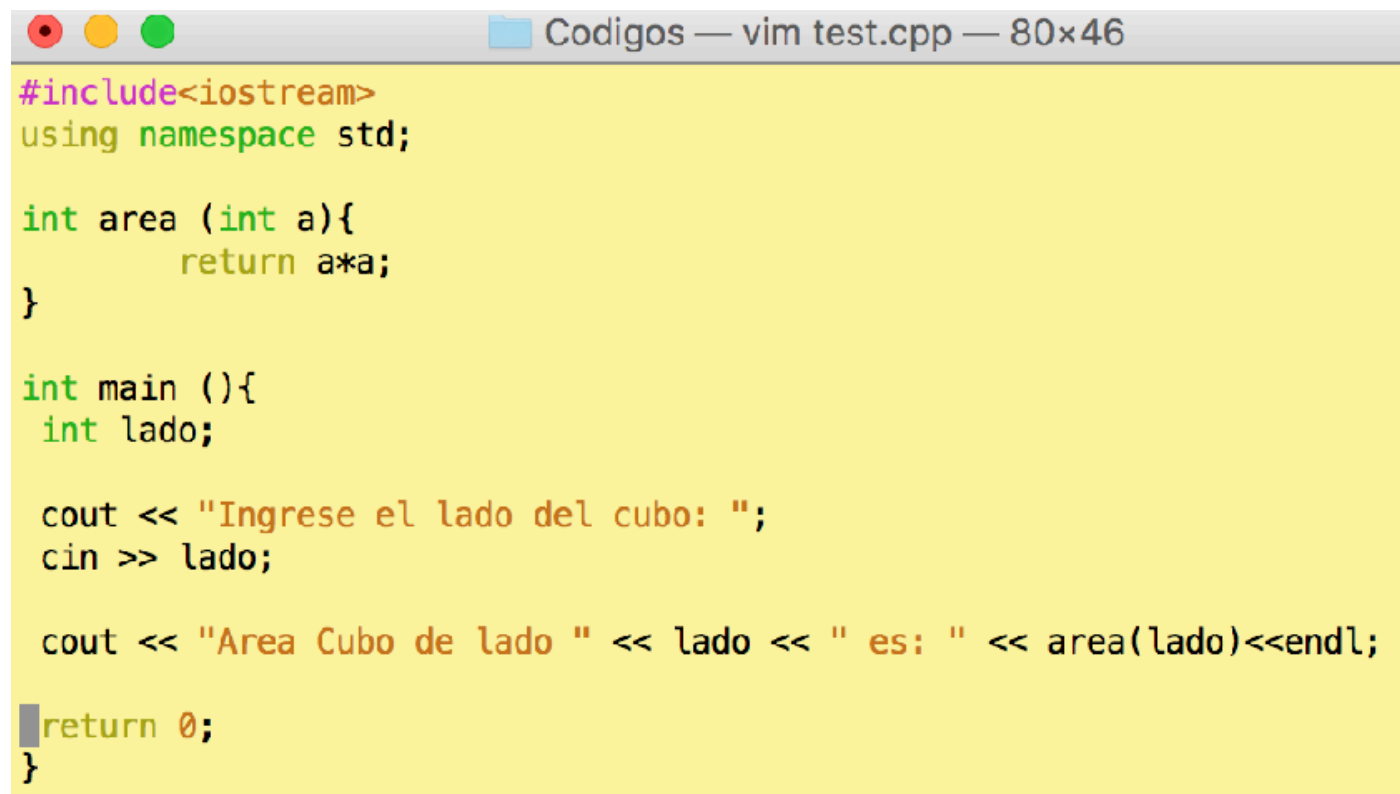
    return 0;
}
```

Funciones

- ¿Puedo usar una función dentro de otra función?

Funciones

- ¿Puedo usar una función dentro de otra función? SI!



```
Codigos — vim test.cpp — 80x46
#include<iostream>
using namespace std;

int area (int a){
    return a*a;
}

int main (){
    int lado;

    cout << "Ingrese el lado del cubo: ";
    cin >> lado;

    cout << "Area Cubo de lado " << lado << " es: " << area(lado)<<endl;

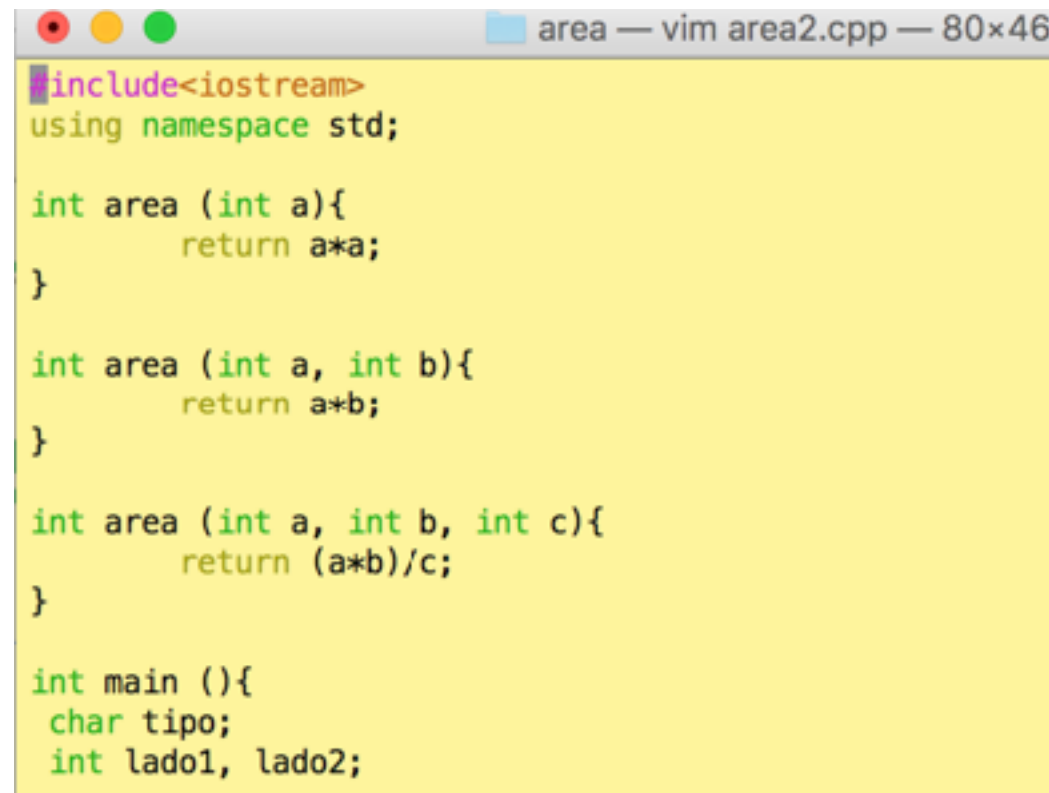
    return 0;
}
```


Funciones

- ¿Será válido definir dos funciones con el mismo nombre?

Funciones

- ¿Será válido definir dos funciones con el mismo nombre? SI
- Funciones sobrecargadas!
- C++, a diferencia del C estándar, permite declarar funciones con el mismo nombre y a esto se conoce como sobrecarga de funciones



```
area — vim area2.cpp — 80x46
#include<iostream>
using namespace std;

int area (int a){
    return a*a;
}

int area (int a, int b){
    return a*b;
}

int area (int a, int b, int c){
    return (a*b)/c;
}

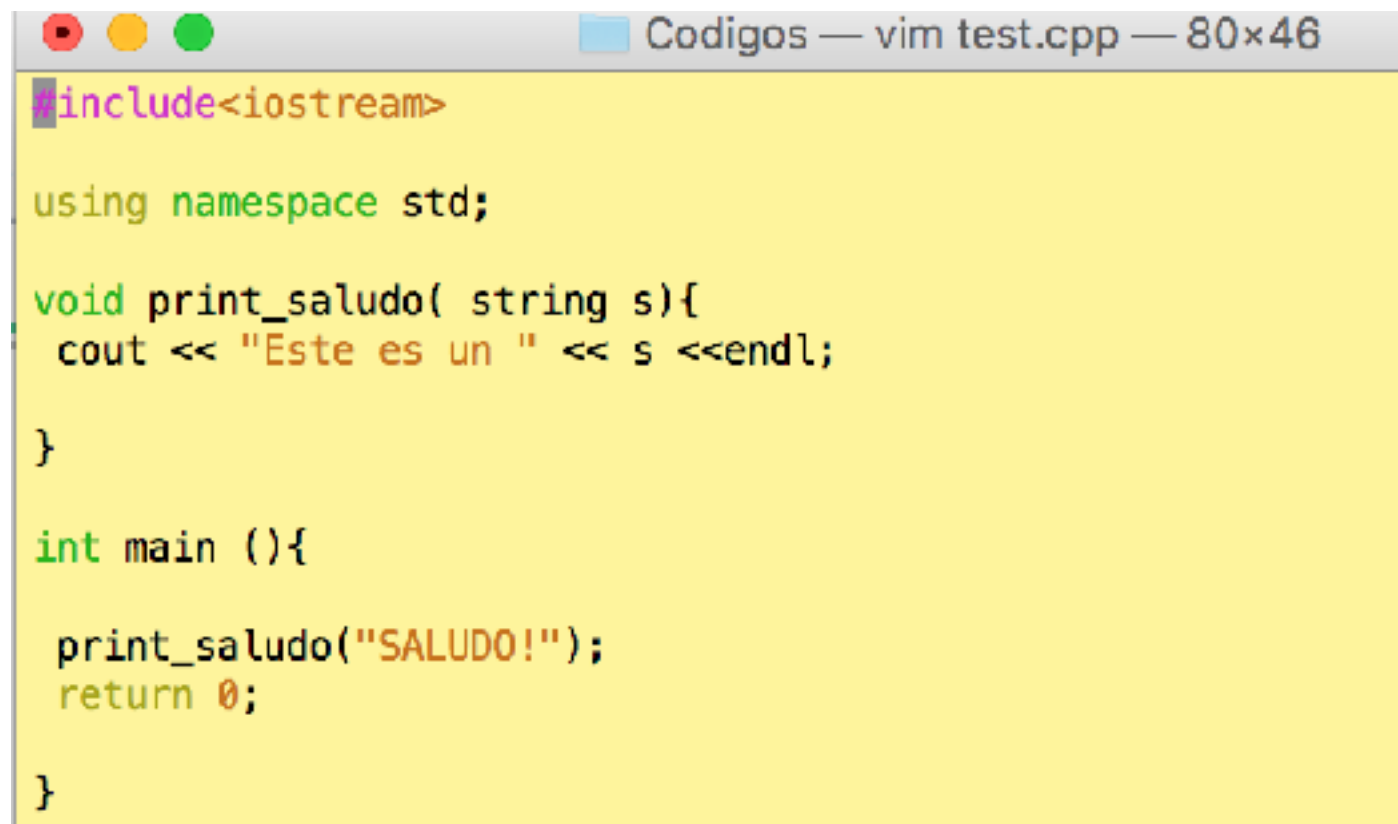
int main (){
    char tipo;
    int lado1, lado2;
```

Funciones

- ¿Podemos tener funciones que no retornen nada?

Funciones

- ¿Podemos tener funciones que no retornen nada? Si!



```
Codigos — vim test.cpp — 80x46
#include<iostream>

using namespace std;

void print_saludo( string s){
    cout << "Este es un " << s <<endl;
}

int main (){

    print_saludo("SALUDO!");
    return 0;
}
```

Ejercicios

- **Ejercicio 1:** escriba un programa en c++ utilizando funciones que permita calcular el área de un rectángulo o cuadrado.

Ejercicios

- **Ejercicio 1.1:** escriba un programa en c++ utilizando funciones que permita calcular el área total de un departamento cuando se ingresan las medidas interiores de cada habitación. Considere que el departamento tiene 2 dormitorios, dos baños y un living-comedor y todos ellos son rectangulares.

Ejercicios

- **Ejercicio 2:** escriba un programa que le permita calcular el factorial de un número ingresado. Cree la función factorial() la cual recibe un entero y retorna en un entero el factorial del número ingresado. Factorial se define como:
 - $\text{factorial}(3) = 1 * 2 * 3 = 6$

Ejercicios

- **Ejercicio 2.2:** Escriba un programa que le permita calcular la combinatoria de dos numero m y n. La combinatoria esta definida como:

$$C(m,n) = m! / (m-n)! * n!$$

Ejercicios

- **Ejercicio 3:** escriba un programa en c++ que permita transformar en hora, minutos y segundos un entero positivo ingresado que se representa en segundos. De solución generando usando las funciones getHoras(), getMinutos(), getSegundos().
- Ejemplo:
 - Ingrese su tiempo en segundos: 9814
 - 2 h 43 min 34 seg

Ejercicios

- **Ejercicio 4:** Se desea realizar un sistema de apuestas donde el usuario ingresa el monto a apostar en dólares, a continuación selecciona 1 cuadrante: a, b, c o d. Una vez realizada la elección y fijado el monto a apostar se procede a lanzar una moneda. Si la moneda cae en el cuadrante seleccionado por el usuario este se lleva el doble del dinero apostado, en caso contrario, pierde lo apostado. Realice un programa que permita realizar este mecanismo de apuesta y notifique al usuario si gana, o pierde mostrando el monto ganado o perdido según el caso. Cree las funciones lanzar() que genera un lanzamiento aleatorio y la función dinero_ganado() que retorna el monto que se le entrega al usuario luego de jugar.

Ingrese su Apuesta: 200

Ingrese el cuadrante que desea apostar (a, b, c o d): b

- Cayó en el cuadrante que seleccionó, Ud ha ganado! se lleva US400!

- Cayó en el cuadrante a! Ud ha perdido US200!

Tips:

Librería : `#include<cstdlib>`;

random: `srand(time(NULL)); double random = rand()%100 +1 (random entre 1 y 100)`

Ejercicios

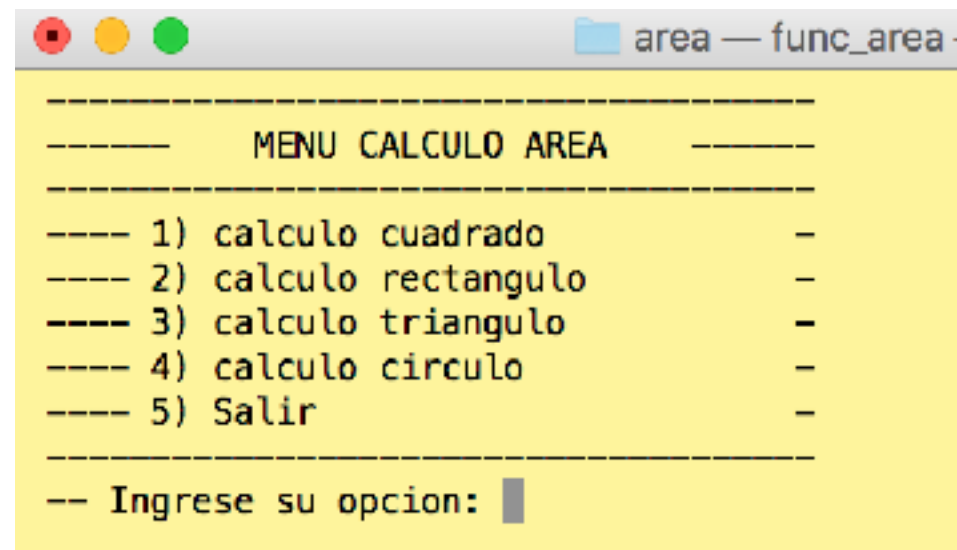
- **Ejercicio 5:** Genere un programa en C++ haciendo uso de funciones, el cual reciba dos números y discrimine cuál de los dos es el mayor. Para ello cree una función que evalúe cuál de los números es mayor.

Ejercicio

- Cree un programa en C++ que dado un número entero de entrada, cuente cuántos dígitos del número son pares, y cuántos son impares. Para ello genere al menos dos funciones:
 - *cuentaPar()* y *cuentaImpar()*

Ejercicios

- **Ejercicio 6:** Genere un programa en C++ haciendo uso de funciones, el cual permita calcular el área de un: triángulo, cuadrado, rectángulo y círculo. Defina una función área capaz de llevar a cabo el cálculo de las áreas (sobrecargar función área). Además genere una función capaz de manejar el menú de opciones a desplegar.



```
-----  
----- MENU CALCULO AREA -----  
-----  
---- 1) calculo cuadrado          -  
---- 2) calculo rectangulo        -  
---- 3) calculo triangulo         -  
---- 4) calculo circulo           -  
---- 5) Salir                     -  
-----  
-- Ingrese su opcion: █
```

Ejercicio

- **Ejercicio 14:** La secuencia de Collatz de un número entero se construye de la siguiente forma:
 - si el número es par, se lo divide por dos;
 - si es impar, se le multiplica tres y se le suma uno;
 - la sucesión termina al llegar a uno.
- Desarrolle un programa que entregue como salida la secuencia de Collatz para un número n ingresado por el usuario. Para ello cree la función `numeroCollatz()`, `esPar()` o `esImpar()`.

n : 18

18 9 28 14 7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

Ejercicio

- Un conocido matemático descubrió un conjunto de números enteros que denominó como los meta-números. Un meta-número es número compuesto por **al menos 3** cifras **primas seguidas**. Por ende un meta-número nunca puede tener menos de 3 cifras. Escriba un programa que permita a cualquier mortal conocer si el número ingresado es o no un meta-número. Resuelva utilizando al menos las funciones: ***esPrimo()***, ***esMetanumero()***.
 - ***E.g: Ingrese un número: 135790***
 - *Es un meta-numero!*

Ejercicio

- Dada la siguiente función:

$$f(x) = -3x^2 + 100x + 30$$

- Diseñe una función (en C++) que retorne el valor real de la función considerando el argumento x como un número entero.
- Implemente un main en donde haciendo uso de la función, encuentre el mayor valor que puede retornar considerando que x puede tomar valores enteros entre 0 y 50.

Ejercicio

- Genere un programa que reciba una palabra y cuente el número de vocales que esta contiene. Para ello, cree la función contarVocales().
 - Ingrese palabra: Hola
 - La palabra tiene 2 vocales!

Ejercicio

- Usando funciones, determinar la cantidad de dinero que debe recibir cada familia por concepto de becas. El monto total de las becas se calcula de la siguiente manera:
 - Solo reciben beca si los ingresos totales son menores a 500.000
 - Por cada hijo entre 6 y 24 años la familia recibe 66.000
 - Los hijos menores de 10 años reciben un monto extra de 30.000
 - Los hijos mayores de edad, reciben un extra de movilización de 17.000
- *Entrada: ingresos familia y edades de los miembros.*

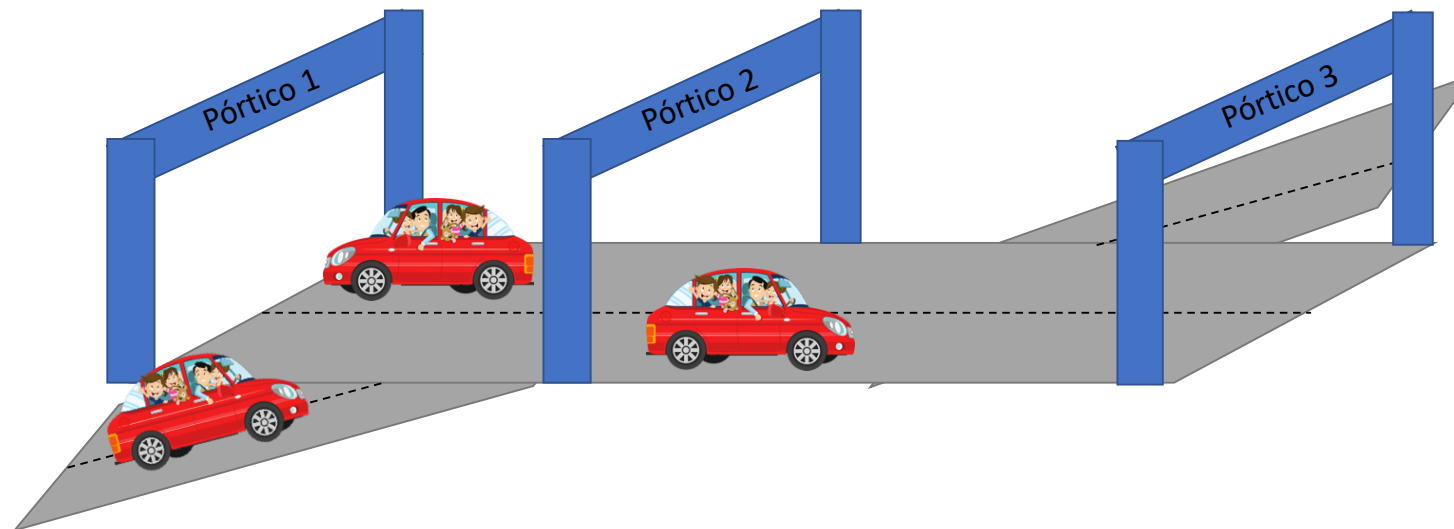
P3.-) (20 Puntos Total)

Una pequeña empresa concesionaria de autopista mantiene tres grandes pórticos de peajes en una ruta con pistas en dos direcciones (Norte-Sur y Sur-Norte). Cada pórtico se encuentra numerado (1,2,3) y estratégicamente solo existen entradas a la autopista en el segmento que cubren los tres pórticos, las entradas se encuentran justo antes del pórtico 2 según el sentido de la vía. Luego de realizar completos estudios de transporte se fijó una tarifa por pórtico de \$4321 pesos en dirección Norte y \$1234 en dirección Sur.

La empresa mantiene un registro por cada vehículo que atraviesa la autopista con la siguiente forma:

- 1 2 3 -> El vehículo paso por el pórtico 1, luego el 2, luego el 3. En dirección Norte.
- 3 2 1 -> El vehículo paso por el pórtico 3, luego el 2, luego el 1. En dirección Sur.
- 0 2 3 -> El vehículo se incorporo en el pórtico 2, y luego paso por el 3. En dirección Norte.
- 0 2 1 -> El vehículo se incorporo en el pórtico 2, y luego paso por el 1. En dirección Sur.

Cualquier registro fuera de orden implica una falla en el sistema de registro producto de evasión (patente oculta).



A usted se le pide un programa que lea los registros de 1000 vehículos e indique lo siguiente:

1. Cantidad de vehículos en dirección Norte.
2. Cantidad de vehículos en dirección Sur.
3. Total ingresos por vehículos en dirección Norte.
4. Total ingresos por vehículos en dirección Sur.
5. Cantidad de evasiones.

Funciones Recursivas

Funciones Recursivas

- Se dice que una función es recursiva cuando se define en función de sí misma.
- No todas la funciones pueden llamarse a si mismas, sino que deben estar diseñadas especialmente para que sean recursivas, de otro modo podrían conducir a bucles infinitos, o a que el programa termine inadecuadamente
- Tampoco todos los lenguajes de programación permiten usar recursividad.

Funciones Recursivas

- Dentro de una función recursiva suelen distinguirse dos partes:
 - **Los casos base:** Son aquellos que para su solución no requieren utilizar la función que se está definiendo.
 - **Los casos recursivos:** Son aquellos que sí que requieren utilizar la función que se está definiendo.
- Las definiciones recursivas funcionan siempre y cuando las llamadas recursivas se realicen de forma que en algún momento se lleguen a los casos base.

Funciones Recursivas

- ¿Qué hace esta función si le pasamos como parámetro $n = 4$?
- ¿Cuál será su resultado?

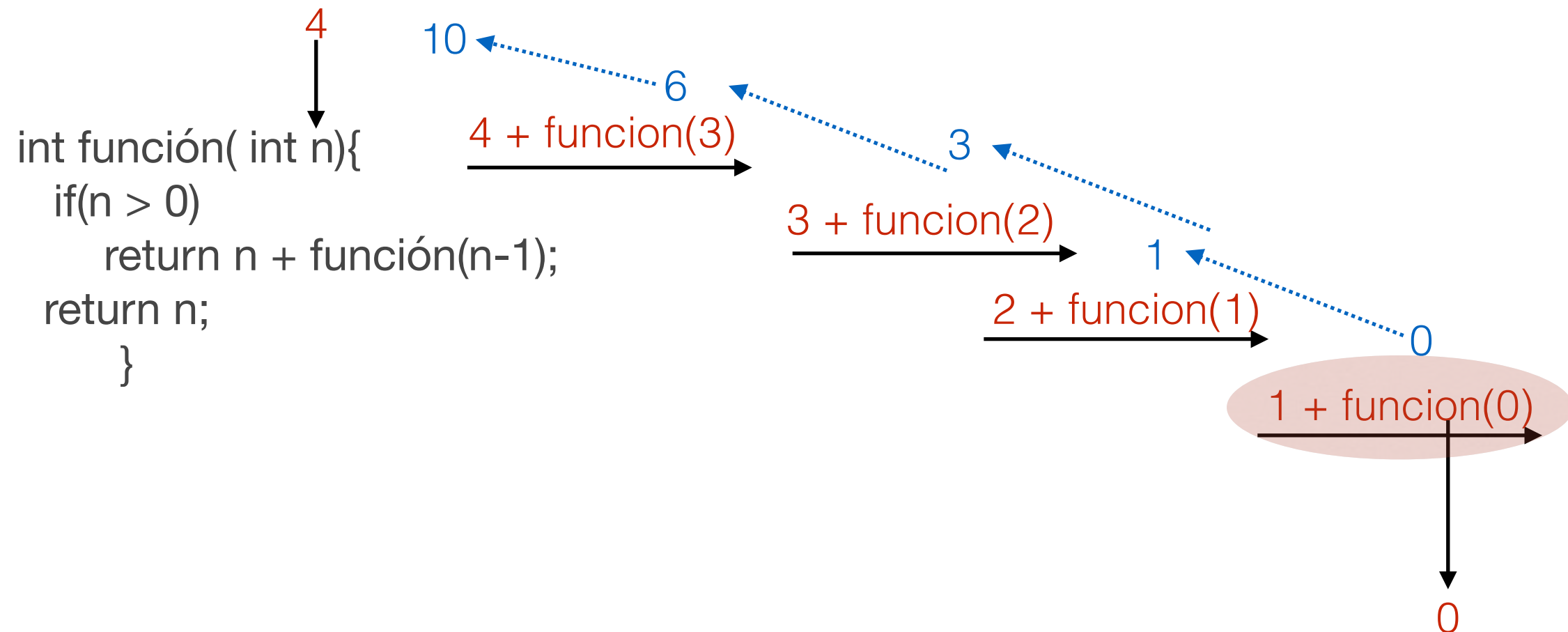
```
int función( int n){  
    if(n > 0)  
        return n + función(n-1);  
    return n;  
}
```

n diferente →

} Llamada recursiva

} Caso base

Función Recursiva



Funciones Recursivas

- **Ejercicio 7:** Genere un programa que calcule el factorial de un número **n** ingresado por consola por el usuario haciendo uso de funciones recursivas.
 - $\text{factorial}(3) = 1 * 2 * 3 = 6$

Funciones Recursivas

- **Ejercicio 8:** Implemente una función recursiva en C++ que imprima por pantalla los valores desde 1 hasta un número introducido por el usuario.

Funciones Recursivas

- **Ejercicio 9:** Genere un programa que calcule los N (N ingresado por el usuario) primeros números de la serie de Fibonacci haciendo uso de funciones recursivas. Fibonacci es una serie infinita donde se calculan los N primeros términos para la serie y el valor de cada termino se define así.
- $\text{fibonacci}(i) = \text{fibonacci}(i-1) + \text{fibonacci}(i-2)$
- considere que $\text{fibonacci}(0) = 0$ y $\text{fibonacci}(1) = 1$
 - Ejemplo: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55...

Funciones Recursivas

- **Ejercicio 10:** Genere un programa que implemente una función recursiva para invertir un número ingresado por el usuario.
- **Ejemplo:** Entrada: 123 Salida: 321

Funciones Recursivas

- **Ejercicio 11:** Genere un programa que implemente una función recursiva que permita transformar un número expresado en notación decimal a un número binario.