



Universidad de Costa Rica  
Facultad de Ingeniería  
Escuela de Ingeniería Eléctrica  
**IE-0117 Programación Bajo Plataformas Abiertas**



Ing. Marco Villalta Fallas - I Ciclo 2018

---

Laboratorio #5  
Funciones de C

---

**Instrucciones Generales:**

Los laboratorios se deben de realizar de manera individual.

El laboratorio debe de entregarse antes del 10 de mayo a las 23:59.

Utilice capturas de pantalla para demostrar la funcionalidad.

Entregue un archivo comprimido que incluya un directorio llamado **informe** con los archivos necesarios para generar el PDF del informe (.tex, imágenes, código, entre otros) y un directorio llamado **src** con los archivos de código fuente que lleven a la solución. Cualquier otro formato o entrega tardía no se revisará y el laboratorio tendrá una nota de cero.

Documente e investigue los algoritmos solicitados, en el reporte final se debe incluir un diagrama de flujo de la implementación realizada. Cualquier programa que no se compile correctamente no tendrá puntaje. En la calificación de este laboratorio se tomará en cuenta el uso del control de versiones git y la documentación en Doxygen

## 1. Método de la burbuja

Realice una función en C que ordene descendentemente un arreglo de N elementos utilizando el método de la burbuja. Debe solicitar al usuario el arreglo (máximo de 10 elementos).

## 2. Suma de números pares

Realice una función en C que realice la suma de los números pares enteros entre un límite inferior **a** y un límite superior **b**, debe regresar la suma en la variable **c**. Los límites deben ser solicitados al usuario. Realice un programa principal que compruebe su funcionalidad.

## 3. Solución de una ecuación cuadrática

Realice una función en C para encontrar las dos soluciones reales de una ecuación cuadrática que tiene la forma:

$$ax^2 + bx + c = 0 \quad (1)$$

Si la ecuación no tiene soluciones reales, debe imprimir en pantalla que no tiene soluciones reales. Si las tiene, debe imprimir ambas soluciones. Realice un programa principal que compruebe su funcionalidad. (Puntos extra: mostrar soluciones no reales)

## 4. Torre de Hanoi

Implemente un programa que solicite la cantidad de discos iniciales y muestre paso a paso la solución, donde se debe utilizar una función de forma recursiva.

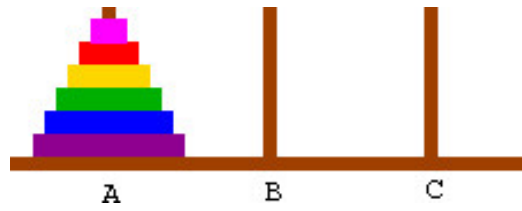


Figura 1: Representación de la torre de hanoi

De referencia, el pseudo-algoritmo es:

1. Mover n-1 discos de la clavija fuente a la clavija auxiliar.
2. Mover el disco n de la clavija fuente a la clavija destino
3. Mover el n-1 discos de la clavija auxiliar a la clavija destino

```

FUNCTION MoveTower(disk, source, dest, spare):
IF disk == 0, THEN:
    move disk from source to dest
ELSE:
    MoveTower(disk - 1, source, spare, dest)
    move disk from source to dest
    MoveTower(disk - 1, spare, dest, source)
END IF

```