

## Calentamiento de una placa por el método de Gauss-Seidel

### Paso 1. Crear archivos con los programas.

*pmain.c* - Función principal del programa.

*funciones.c* - Definición de todas las funciones utilizadas en el programa.

*graficas.py* - Script en Python para hacer las gráficas.

### Paso 2. Iniciar variables.

En *main.c*:

*Tipo puntero:*

float placa = Puntero doble que asigna la memoria utilizada por cada estado de la placa.

float PlacaVieja = Puntero doble que asigna la memoria utilizada por cada estado previo de la placa.

char argv = Puntero simple para guardar los argumentos dados en la línea de comandos.

*Tipo entero:*

argc = Número de argumentos recibidos en la línea de comandos.

max\_iter = 1000 = número máximo de iteraciones del método.

Iter = 1 = contador de iteraciones.

Cumple = 0 = entero que cumple la función de un booleano, usado para señalar si ya se alcanzó el equilibrio en la placa y, si es el caso, detener las iteraciones antes de alcanzar el máximo.

n = Medida de la placa.

### Paso 3. Leer argv.

argv contendrá las temperaturas de los bordes de la placa y 'n', el número de puntos interiores.

### Paso 4. Calcular la medida del lado de la placa a partir de n.

Asignar a n este nuevo valor.

### Paso 5. Iniciar placa y placaVieja.

placa será un arreglo 2D de números flotantes representando la temperatura de la placa en el punto correspondiente a los índices de ese flotante. Inicialmente, las temperaturas de los bordes serán dadas por el usuario y las interiores serán cero.

### Paso 6. Crear archivo con placa inicial.

**Paso 7. Ciclo while (iter<=max\_iter && cumple==0)**

- i. Aplicar el método de Gauss en cada punto de placa.
- ii. Verificar si la placa ya alcanzó el equilibrio.
- iii. Copiar placa actual a placaVieja.
- iv. Crear un archivo con la placa actual. Cada archivo se guardará en el directorio 'archivos'.
- v. iter++

**Paso 8. Liberar la memoria.**

**Paso 9. Abrir graficas.py**

El script se encuentra en el directorio 'archivos'.

Sea N = número de archivos creados con el código en C.

**Paso 10. Crear una gráfica con cada archivo.**

Cada gráfica se guardará en el directorio 'archivos'.

**Paso 11. Hacer un gif a partir de la unión de todas las gráficas.**