

Proyecto 1: Open MP  
**Catálogo de funciones**

María Isabel Ortiz Naranjo y Luis Alejandro Urbina Hernández

---

## Funciones en archivo pdeSecuencial.cpp

### calcularHeat

Entrada:

- intervalosDistancia: cantidad de intervalos a evaluar (entero)
- intervalosTiempo: cantidad de intervalos de tiempo a evaluar (entero)
- Tl: Temperatura en la frontera izquierda (double)
- Tr: temperatura en la frontera derecha (double)

Descripción: Inicializar la matriz, entrar en un ciclo while desde 1 hasta la cantidad de intervalos de distancia y dependiendo del índice en el que se encuentra el ciclo, asignar a la matriz la temperatura en la frontera izquierda, la temperatura en la frontera derecha, calcular la nueva temperatura  $T_j(t_{i+1})$  y asignarla a la matriz, repetir este proceso hasta que lleguemos a 10,000 iteraciones.

Output: La función es de tipo void pero se puede modificar para mostrar la matriz

### main

Entrada:

- intervalosDistancia: cantidad de intervalos a evaluar (entero)
- Tl: Temperatura en la frontera izquierda (double)
- Tr: temperatura en la frontera derecha (double)

Descripción: Solicitar los datos de entrada y verificar que ninguno sea menor o igual a 0, además validar que la diferencia entre las temperaturas sea mayor a 60,

mostrar los parámetros con los que se está corriendo la función, calcular el tiempo actual, llamar a la función calcularHeat y luego calcular el tiempo de nuevo. Restar el tiempo final menos el inicial y retornar el tiempo secuencial.

Output: La función retorna un int si la ejecución fue correcta.

## Funciones en archivo pdeParalela.cpp

### calcularHeatParalelo

Entrada:

- index: identificador del thread actual
- intervalosDistancia: cantidad de intervalos a evaluar (entero)
- Tl: Temperatura en la frontera izquierda (double)
- Tr: temperatura en la frontera derecha (double)
- myMatriz: matriz que guarda los resultados (Matriz)

Descripción: Declarar las variables compartidas haciendo uso de Open Mp, declarar el tipo de scheduling de threads, entrar en un ciclo while desde 1 hasta la cantidad de intervalos de distancia (esto es una omp task). Dependiendo del índice en el que se encuentra el ciclo, asignar a la matriz la temperatura en la frontera izquierda, la temperatura en la frontera derecha, calcular la nueva temperatura  $T_j(t_{i+1})$  y asignarla a la matriz, luego llamar a la función de nuevo con un índice mayor al actual para el id del thread.

Output: La función es de tipo void pero se puede modificar para mostrar la matriz

### main

Entrada:

- intervalosDistancia: cantidad de intervalos a evaluar (entero)
- Tl: Temperatura en la frontera izquierda (double)
- Tr: temperatura en la frontera derecha (double)

Descripción: Solicitar los datos de entrada y verificar que ninguno sea menor o igual a 0, además validar que la diferencia entre las temperaturas sea mayor a 60,

mostrar los parámetros con los que se está corriendo la función, calcular el tiempo actual, llamar a la función `calcularHeatParalelo` con una matriz construida por los `intervalosDistancia` X `intervalosTiempo` y luego calcular el tiempo de nuevo. Restar el tiempo final menos el inicial y retornar el tiempo secuencial.

Output: La función retorna un int si la ejecución fue correcta.