Step 1: Problem Identification and Statement

This assignment aims to develop software that simulates the temperature distribution in a thin metal plate with constant (isothermal) temperatures on each side associated with the thermal stimulation, using a 2D grid with unknown dimensions. The program presents the user with a menu to create the dynamic grid, set initial side temperatures, stimulate the grid at a specific point, calculate the thermal distribution, and print the grid on the screen.

Step 2: Gathering of Information and Input/Output Description

Relevant information:

Analyzing the thermal distribution in a thin metal plate is crucial for various engineering and industrial applications. Thermal distribution refers to the way heat is dispersed across the surface of the plate, and understanding this distribution is essential for Heat Management, Material Integrity, Optimizing the performance of materials, Energy efficiency of the material, etc.

With the virtual tool for Thermal distribution, the code allows engineers and researchers to analyze and understand the heat behavior of materials, being practical in different areas such as Electronics and Microelectronics, Aerospace Engineering, Energy systems, Material Science, etc.

The thermal distribution is calculated through the arithmetic mean of the temperatures of the four points around the point at which the temperature will change - a simulation of the temperature induction process.

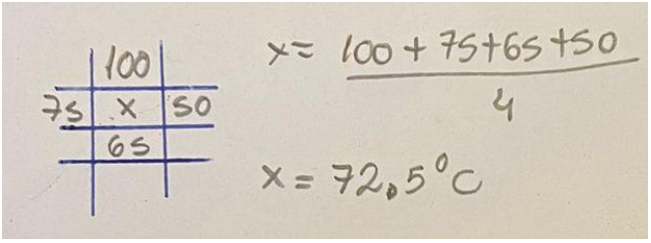
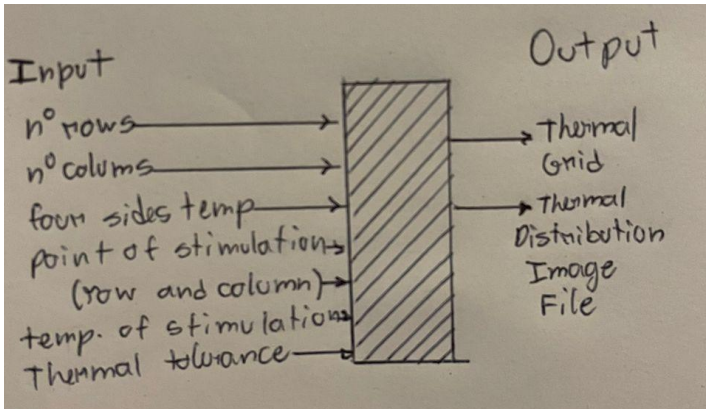


Figure 1: Representation of the thermal distribution

Input/output Description:



The program requires 5 inputs from the user—the number of rows and columns on Option 1 to create the grid; Option 2, the four sides' temperature, option 3, the temperature of the stimulation point and the localization of the point (column and row), and on the Option 4 the thermal tolerance. The Output is the Thermal Grid and the Thermal Distribution Image File - from the Header File with the temperatures of the code.

Step 3: Design of the algorithm and test cases

Test Case 1: Invalid Menu Option:

```
Select one of the options below to simulate the thermal distribution
in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific
stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above: 7
Invalid option, try again.
Select one of the options of the main menu above
```

Test Case 2: Going to another step without creating the grid first.

```
Select one of the options below to simulate the thermal distribution
in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific
stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above:2
The creation of the grid is needed for this step
Select one of the options of the main menu above:
```

Test Case 3: Choosing a point of stimulation outside the size of the grid:

```
Select one of the options below to simulate the thermal distribution
in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
```

```
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific
stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above: 1
Inform the number of rows: 4
Inform the number of colums: 4
Grid created
Select one of the options of the main menu above 2
Please inform the initial temperature of the top of the grid 50
Please inform the initial temperature of the bottom of the grid 50
Please inform the initial temperature of the left of the grid 30
Please inform the initial temperature of the right of the grid 30
Temperature setted
Select one of the options of the main menu above 3
Please inform the row of the grid where do you want to make the
stimulation 8
Please inform the column of the grid where do you want to make the
stimulation 10
Please inform the temperature value of the stimulation 120
Invalid option, please select the values again
Please inform the row of the grid where do you want to make the
stimulation
```

Test Case 4: Checking Image output of the thermal distribution

```
Select one of the options below to simulate the thermal distribution
in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific
stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above
1
Inform the number of rows:
9
Inform the number of columns:
9
Grid created
Select one of the options of the main menu above
2
```

```
Please inform the initial temperature of the top of the grid
70
Please inform the initial temperature of the bottom of the grid
90
Please inform the initial temperature of the left of the grid
20
Please inform the initial temperature of the right of the grid
30
Temperature setted
Select one of the options of the main menu above
3
Please inform the row of the grid where do you want to make the
stimulation
5
Please inform the column of the grid where do you want to make the
stimulation
6
Please inform the temperature value of the stimulation
272
70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    272.00  0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00
Select one of the options of the main menu above
4
Enter the simulation tolerance (a positive value)3
70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00
20.00   41.70   49.19   52.35   53.96   54.52   53.32   47.82   30.00
20.00   29.79   35.82   39.52   41.91   43.12   42.47   38.57   30.00
20.00   24.81   29.10   32.57   35.27   36.98   37.08   34.80   30.00
20.00   24.11   28.26   31.87   34.75   36.57   36.69   34.47   30.00
20.00   27.31   33.31   37.69   40.67   42.09   41.25   37.26   30.00
20.00   35.37   44.71   49.97   52.76   53.42   51.12   43.96   30.00
20.00   51.95   63.39   68.04   69.95   69.96   67.13   57.77   30.00
90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00
Thermal distribution saved as an image.
Select one of the options of the main menu above
6
exited the program
```

(The grid are well separated and the thermal distribution is well made on the program)

Test Case 5: Choosing an invalid tolerance value (other value than a number)

```
Select one of the options below to simulate the thermal distribution
in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific
stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above 1
Inform the number of rows: 4
Inform the number of colums: 4
Grid created
Select one of the options of the main menu above 2
Please inform the initial temperature of the top of the grid 50
Please inform the initial temperature of the bottom of the grid 50
Please inform the initial temperature of the left of the grid 30
Please inform the initial temperature of the right of the grid 30
Temperature setted
Select one of the options of the main menu above 3
Please inform the row of the grid where do you want to make the
stimulation 2
Please inform the column of the grid where do you want to make the
stimulation 1
Please inform the temperature value of the stimulation
100
50.00   50.00   50.00   50.00
30.00   0.00    0.00    30.00
30.00   100.00  0.00    30.00
50.00   50.00   50.00   50.00
Select one of the options of the main menu above 4
Enter the simulation tolerance (a positive value)b
Invalid input, enter a number:
```

Algorithm design:

Let's separate functions from the main code:

Functions:

Function create2DGrid(rows,cols):

Assign a new array of pointers with size rows to grid

```
            Repeat with i from 0 to rows - 1:
                Assign grid[i] to a new array of doubles with size cols

        Repeat with i from 0 to rows -1
            Repeat with j from 0 to cols - 1
                Assign 0 to grid[i][j]
        Return ptr

Function setsideTemperature(grid, rows, cols):
(get values from the user)
        Assign temp (4)
            Print "Please inform the initial temperature of the top of the grid"
            Read temp(0) from the user
            Print "Please inform the initial temperature of the bottom of the grid"
            Read temp(1) from the user
            Print "Please inform the initial temperature of the left of the grid"
            Read temp(2) from the user
            Print "Please inform the initial temperature of the right of the grid"
            Read temp(3) from the user (setting the temperatures for each row and column)
            Assign grid(0)(i) to temp(0) for each i from 0 to cols - 1
            Assign grid (rows-1)(i) to temp(1) for each i from 0 to cols - 1
            Assign grid(j)(0) to temp(2) for each j from 1 to rows - 2
            Assign grid(j)(cols-1) to temp(3) for each j from 1 to rows - 2
            (Function for specific grid point stimulation)

Function stimulateGrid(grid, rows, cols):
        Assign a, b, temp (from the user)
        (Prompting the user to stimulate the grid)
        Print "Please inform the row of the grid where do you want to make the stimulation"
        Read into a
        Print "Please inform the column of the grid where do you want to make the stimulation"
        Read into b
        Print "Please inform the temperature value of the stimulation"
        Read into temp

        // Invalid choice case If b is less than 1 OR b is greater than cols OR a is less than 1 OR
        a is greater than rows
        Print "Invalid choice"
        While (b is greater than 1 and b is less than cols anda is greater than 1 and a is less
        than rows)
        Assign grid(a)(b) to temp

        Function displayGrid (grid, rows, cols):
            Set precision to 2
```

```
                For (i from 0 to rows - 1) ( Iterate over columns )
                        For (j from 0 to cols - 1)
                                Print grid (i, j) ( Move to the next line after printing each
                        row ) Print a new line

Function calculateTD(grid, rows, cols, tolerance):
        Assign thermEquilibrium to false
                Repeat
                Assign maxchange to 0
                        Repeat with i from 1 to rows - 2
                                Repeat with j from 1 to cols - 2
                        Assign temp to the average of (grid(i)(j + 1), grid(i)(j - 1), grid(i -
                        1)(j), grid(i +
        1)(j)) divided by 4
                        Assign chng to the absolute value of (grid(i)(j) - temp)
                        If (chng is greater than maxchange)
                                Assign maxchange to chng
                        Assign grid(i)(j) to temp

        If (maxchange is less than or equal to tolerance)
                Assign thermEquilibrium to true
        (Ensure temperatures are within a valid range (0 to 255) for image conversion)
        Repeat with i from 0 to rows - 1
                Repeat with j from 0 to cols - 1
                        If grid(i)(j) is less than 0
                                Assign grid(i)(j) to 0
                        Otherwise if grid(i)(j) is greater than 255:
                                Assign grid(i)(j) to 255
                Repeat while thermEquilibrium is false

        Function deleteGrid(grid, rows):
                Repeat with i from 0 to rows - 1:
                        Deallocate memory for grid[i]
                        Delete grid

------------------ Start of Main Function -----------
Assign row = 0 (Number of rows in the 2D grid)
Assign col = 0 (Number of columns in the 2D grid)
Assign input (User's menu selection)
Assign toler = 0 (Tolerance value for thermal distribution)
Assign grid to nullptr

Print "Select one of the options below to simulate the thermal distribution in a thin metal
plate with constant(isothermal)"
```

```
Print "1) Create the dynamic 2D grid."
Print "2) Set initial side temperature values for the grid."
Print "3) Stimulate the grid at a specific point."
Print "4) Calculate the thermal distribution based on the specific stimulation and initial conditions."
Print "5) Print the thermal grid on the screen."
Print "6) Exit the program."

Repeat
    Print "Select one of the options of the main menu above "
    While (input is not a valid value)
        Print "Invalid option, please try again: "
        Clear input error flag
        Ignore invalid input


Switch option (present the cases)
    Case 1:
        Print "Enter the number of rows: "
        Read into row
        Print "Enter the number of columns: "
        Read into col
        Assign arr to create2DGrid(row, col) (Create a dynamic 2D grid)
        Print”Grid Created”
Case 2:
If (arr is equal to nullptr) (If the grid was not created)
    Print "The creation of the grid is needed for this step"
Otherwise
    Call the function setsideTemperature with parameters (arr, row, col)
    Print “Temperature setted”

Case 3:
    If (arr is equal to nullptr) (If the grid was not created)
        Print "The creation of the grid is needed for this step"
    Otherwise
    Call the function stimulateGrid with parameters (arr, row, col)
    Call the function displayGrid with parameters(arr, row, col)

Case 4:
    If (array is equal to nullptr) (If the grid was not created)
        Print "The creation of the grid is needed for this step"
    Otherwise
        Print "Enter the stimulation tolerance (a positive value): "
        While (tolerance input is not a valid number)
```

Print "Invalid input, enter a valid number: "
Clear input error flag
Ignore invalid input
Call function simulateThermalDissipation with parameters (arr, row, col, toler) (Calculate thermal distribution)
(Save thermal distribution as an image)
Call function displayGrid with parameters (arr, row, col)

Assign filename to "Outputimage.bmp"
Assign image to a 2D uint8_t (rows)
Repeat for each row from 0 to rows - 1
Assign image[row] to uint8_t (cols)
Repeat for each column from 0 to cols - 1
Assign image(row)(column) to uint8_t value converted from arr(row)(column)

Call function writeBitmap with parameters (filename, image, col, row);
Print "Thermal distribution saved as an image."

For each value of i from 0 to rows - 1
Deallocate memory for the image array
Delete image array

Case 5:
If (array is equal to nullptr) (If the grid was not created)
Print "The creation of the grid is needed for this step"
Otherwise
Call the function displayGrid with parameters (arr, row, col) (Print thermal grid on the screen)

Case 6:
Display "Exiting the program"
Call the function DeleteArray with parameters (array, rows) (Deallocate memory)
Return 0
Default:
Display "Invalid option, try again”
End Switch

Step 4: Implementation

```
/*************************************************************
Author: Pedro Felix Fernandes
Date Created: November 10, 2023
Description:

_____________________________________________________________
```

Assignment 2 - Mechanical Engineering Case Study – Thermal Distribution Simulation

_____

```
Mechanical Engineering Case Study – Thermal Distribution Simulation:
The program simulates the temperature distribution in a thin metal
plate with constant (isothermal) temperatures
on each side associated with the thermal stimulation, using a 2D grid
with unknown dimensions (number of rows and columns).
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific
stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
************************************************/

// Including Libraries and Header Files
#include <iostream>
#include <cmath>
#include <iomanip>
#include "Header.h"
// defining standard namespace
using namespace std;

// functions that will be used on the program
double** create2DGrid(int rows, int cols);
void setsideTemperature(double** grid, int rows, int cols);
void stimulateGrid(double** grid, int rows, int cols);
void displayGrid(double** grid,int rows, int cols);
void simulateThermalDissipation(double** grid, int rows, int cols,
double tolerance);
void deleteGrid(double** grid, int rows);

// main function
int main() {
     // global variables
     double toler = 0;
     int input = 0;
     int row = 0;
     int col = 0;
     double **arr = nullptr;

     // menu options
```

```
     cout << "Select one of the options below to simulate the
thermal distribution in a thin metal plate with constant(isothermal)"
<< endl;
     cout << "1) Create the dynamic 2D grid." << endl;
     cout << "2) Set initial side temperature values for the grid."
<< endl;
     cout << "3) Stimulate the grid at a specific point." << endl;
     cout << "4) Calculate the thermal distribution based on the
specific stimulation and initial conditions." << endl;
     cout << "5) Print the thermal grid on the screen." << endl;
     cout << "6) Exit the program." << endl;

     while (true) {
          cout << "Select one of the options of the main menu above
" << endl;

          while (!(cin >> input)) { // validation of the input
               cin.clear(); //error flag clearing
               cin.ignore(numeric_limits<streamsize>::max(),
'\n');// discard the invalid input
          }
          switch (input) {
               // Option 1 - Create a grid
          case 1:
               cout << "Inform the number of rows: " << endl;
               cin >> row;
               cout << "Inform the number of columns: " << endl;
               cin >> col;

               arr=create2DGrid(row, col); // call function to
create the 2D grid using input values

               cout << "Grid created" << endl; // Creation
confirmation
               break;
          //Option 2 - Set initial side temperature values for the
grid.
          case 2:
               if (arr == nullptr) { // validation if option 1 was
made before
                    cout << "The creation of the grid is needed for
this step" << endl;
               }
               else {
```

```
                    setsideTemperature(arr, row, col); // call
function to set temperature on the sides
                    cout << "Temperature setted" << endl;
               }
               break;
          // Option 3 - Stimulate the grid at a specific point.
          case 3:
               if (arr == nullptr) { // validation if option 1 was
made before
                    cout << "The creation of the grid is needed for
this step" << endl;
               }
               else {
                    stimulateGrid(arr, row, col); // Call function
to stimulate the Grid
                    displayGrid(arr, row, col); // Call function to
print the grid
               }
               break;
          // Option 4 - Calculate the thermal distribution based on
the specific stimulation and initial conditions.
          case 4:
               if (arr == nullptr) { // validation if option 1 was
made before
                    cout << "The creation of the grid is needed for
this step" << endl;
               }
               else {
                    cout << "Enter the simulation tolerance (a
positive value)"; // simulation tolerance (treshold)
                    while (!(cin >> toler)){
                         cout << "Invalid input, enter a number:"
<< endl;
                         cin.clear(); // clear error flag

cin.ignore(numeric_limits<streamsize>::max(), '\n');
                    }
                    simulateThermalDissipation(arr, row, col,
toler);//Call function to make the Thermal Dissipation of the grid
                    displayGrid(arr, row, col); // call the
function to print the grip

                    // Store the resulting thermal distribution
data on an image file.
                    const char* filename = "Outputimage.bmp";
```

```
                    uint8_t** image = new uint8_t * [row];
                    for (int i = 0; i < row; ++i) {
                         image[i] = new uint8_t[col];
                         for (int j = 0; j < col; ++j) {
                              image[i][j] =
static_cast<uint8_t>(arr[i][j]);
                         }
                    }
                    writeBitmap(filename, image, col, row);// Call
function to write bit map of image
                    cout << "Thermal distribution saved as an
image." << endl;

                    // Delete dynamic array of image
                    for (int i = 0; i < row; ++i) {
                         delete[] image[i];
                    }
                    delete[] image;
               }
               break;
               // Option 5 - Print the thermal grid on the screen.
          case 5:
               if (arr == nullptr) { // validation if option 1 was
made before
                    cout << "The creation of the grid is needed for
this step" << endl;
               }
               else {
                    displayGrid(arr, row, col); // Call fucntion to
print grid
               }
               break;
               //Option 6 - Exit the program
          case 6:
               cout << "exited the program" << endl;
               deleteGrid(arr, row); // Call function to delete
dynamic grid
               return 0;
          default:
               cout << "Invalid option, try again" << endl;
               break;

          }
     }
}
```

```
void simulateThermalDissipation(double** grid, int rows, int cols,
double tolerance) // function to simulate thermal dissipation
{
     bool thermEquilibrium = false; // since the thermal equilibrium
wasn't reached, we define as false
     do { // Thermal distribution calculation:
               double maxchange = 0;
               for (int i = 1; i < rows - 1; i++) {
               for (int j = 1; j < cols - 1; j++) {
                    double temp = (grid[i][j + 1]
                         + grid[i][j - 1]
                         + grid[i + 1][j]
                         + grid[i - 1][j])/4;
                    double chng = abs(grid[i][j] - temp);
                    if (chng > maxchange) {
                         maxchange = chng;
                    }
                    grid[i][j] = temp;
               }
          }

          if (maxchange <= tolerance) {
               thermEquilibrium = true; // When the thermal
distribution is reached, the value is true
          }
          // Ensure that temperatures are within a valid range (0 to
255) for image conversion
          for (int i = 0; i < rows; i++) {
               for (int j = 0; j < cols; j++) {
                    if (grid[i][j] < 0) {
                         grid[i][j] = 0;
                    }
                    else if (grid[i][j] > 255) {
                         grid[i][j] = 255;
                    }
               }
          }
     } while (!thermEquilibrium);
}

// Print the Grid
void displayGrid(double** grid, int rows, int cols) {
     cout << fixed << setprecision(2);
     for (int i = 0; i < rows; i++) {
```

```
            for (int j = 0; j < cols; j++) {
                  cout << grid[i][j] << "\t";
            }
            cout << endl;
      }
}

// Stimulate the Grid
void stimulateGrid(double** grid, int rows, int cols) {
      int a, b;
      double temp;
      do { // Ask for the coordinates and value of temperature of the
stimulation
            cout << "Please inform the row of the grid where do you
want to make the stimulation" << endl;
            cin >> a;
            cout << "Please inform the column of the grid where do you
want to make the stimulation" << endl;
            cin >> b;
            cout << "Please inform the temperature value of the
stimulation" << endl;
            cin >> temp;

            // Invalid Option verifier
            if (b<1 || b>cols || a<1 || a>rows) {
                  cout << "Invalid option, please select the values
again" << endl;
            }
      } while (b<1 || b>cols || a<1 || a>rows);
            grid[a][b] = temp;
}

// Set the temepratures of the sides of the grid
void setsideTemperature(double** grid, int rows, int cols) {
      //getting the temperatures from the user
      double temp[4];
      cout << "Please inform the initial temperature of the top of
the grid" << endl;
      cin >> temp[0];
      cout << "Please inform the initial temperature of the bottom of
the grid" << endl;
      cin >> temp[1];
      cout << "Please inform the initial temperature of the left of
the grid" << endl;
      cin >> temp[2];
```

```
	cout << "Please inform the initial temperature of the right of
the grid" << endl;
	cin >> temp[3];

	// insert values on the top
	for (int i = 0; i < cols; i++) {
		grid[0][i] = temp[0];
	}
	// insert the values on the bottom
	for (int i = 0; i < cols; i++) {
		grid[rows-1][i] = temp[1];
	}
	// insert the values on the left
	for (int i = 1; i < rows - 1; i++) {
		grid[i][0] = temp[2];
	}
	// insert the values on the right
	for (int i = 1; i < rows - 1; i++) {
		grid[i][cols-1] = temp[3];
	}

}

// Create 2D grid
double** create2DGrid( int rows, int cols) {
	double** grid = new double* [rows];
	for (int i = 0; i < rows; i++) {
		grid[i] = new double [cols];
	}

	// define all values of the grid to 0, after creating it
	for (int i = 0; i < rows; i++) {
		for (int j = 0; j < cols; j++) {
			grid[i][j] = 0;
		}
	}
	return grid;
}

// Delete the dynamic grid - memory deallocation
void deleteGrid(double** grid, int rows) {
	for (int i = 0; i < rows; ++i) {
		delete[] grid[i];
	}
	delete[] grid;
```

```
}
```

Step 5: Software Testing and Verification

Test Case 1: Invalid Menu Option:

```
Select one of the options below to simulate the thermal distribution in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above
7
Invalid option, try again
Select one of the options of the main menu above
```

Test Case 2: Going to another step without creating the grid first.

```
Select one of the options below to simulate the thermal distribution in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above
2
The creation of the grid is needed for this step
Select one of the options of the main menu above
```

Test Case 3: Choosing a point of stimulation outside the size of the grid.

```
Select one of the options below to simulate the thermal distribution in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above
1
Inform the number of rows:
4
Inform the number of columns:
4
Grid created
Select one of the options of the main menu above
2
Please inform the initial temperature of the top of the grid
50
Please inform the initial temperature of the bottom of the grid
50
Please inform the initial temperature of the left of the grid
30
Please inform the initial temperature of the right of the grid
30
Temperature setted
Select one of the options of the main menu above
3
Please inform the row of the grid where do you want to make the stimulation
8
Please inform the column of the grid where do you want to make the stimulation
10
Please inform the temperature value of the stimulation
120
Invalid option, please select the values again
Please inform the row of the grid where do you want to make the stimulation
```

Test Case 4: Checking Image output of the thermal distribution

```
Select one of the options below to simulate the thermal distribution in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above
1
Inform the number of rows:
9
Inform the number of columns:
9
Grid created
Select one of the options of the main menu above
2
Please inform the initial temperature of the top of the grid
70
Please inform the initial temperature of the bottom of the grid
90
Please inform the initial temperature of the left of the grid
20
Please inform the initial temperature of the right of the grid
30
Temperature setted
Select one of the options of the main menu above
3
Please inform the row of the grid where do you want to make the stimulation
5
Please inform the column of the grid where do you want to make the stimulation
```

```
6
Please inform the temperature value of the stimulation
272
70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    272.00  0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
20.00   0.00    0.00    0.00    0.00    0.00    0.00    0.00    30.00
90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00
Select one of the options of the main menu above
4
Enter the simulation tolerance (a positive value)3
70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00   70.00
20.00   41.70   49.19   52.35   53.96   54.52   53.32   47.82   30.00
20.00   29.79   35.82   39.52   41.91   43.12   42.47   38.57   30.00
20.00   24.81   29.10   32.57   35.27   36.98   37.08   34.80   30.00
20.00   24.11   28.26   31.87   34.75   36.57   36.69   34.47   30.00
20.00   27.31   33.31   37.69   40.67   42.09   41.25   37.26   30.00
20.00   35.37   44.71   49.97   52.76   53.42   51.12   43.96   30.00
20.00   51.95   63.39   68.04   69.95   69.96   67.13   57.77   30.00
90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00   90.00
Thermal distribution saved as an image.
Select one of the options of the main menu above
6
exited the program
```

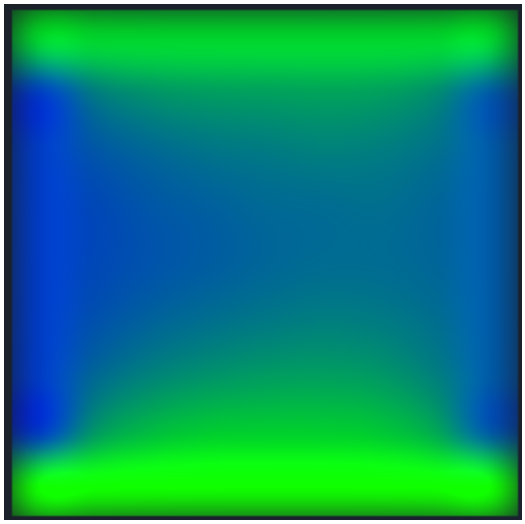


Image that represents the Thermal distribution

Test Case 5: Choosing an invalid tolerance value (other value than a number)

```
Select one of the options below to simulate the thermal distribution in a thin metal plate with constant(isothermal)
1) Create the dynamic 2D grid.
2) Set initial side temperature values for the grid.
3) Stimulate the grid at a specific point.
4) Calculate the thermal distribution based on the specific stimulation and initial conditions.
5) Print the thermal grid on the screen.
6) Exit the program.
Select one of the options of the main menu above
1
Inform the number of rows:
4
Inform the number of columns:
4
Grid created
Select one of the options of the main menu above
2
Please inform the initial temperature of the top of the grid
50
Please inform the initial temperature of the bottom of the grid
50
Please inform the initial temperature of the left of the grid
30
Please inform the initial temperature of the right of the grid
30
Temperature setted
Select one of the options of the main menu above
3
Please inform the row of the grid where do you want to make the stimulation
2
Please inform the column of the grid where do you want to make the stimulation
```

```
Please inform the column of the grid where do you want to make the stimulation
1
Please inform the temperature value of the stimulation
100
50.00   50.00   50.00   50.00
30.00   0.00    0.00    30.00
30.00   100.00  0.00    30.00
50.00   50.00   50.00   50.00
Select one of the options of the main menu above
4
Enter the simulation tolerance (a positive value)b
Invalid input, enter a number:
```

User Guide

This program will help you simulate the temperature distribution of a thin metal plate with constant (isothermal) temperatures on each side with a termal stimulation, on a 2D grid with dimensions informed by the user (row and columns). You will be asked to inform the number of rows and columns of the 2D grid (Option 1), then, after selecting Option 2, you will be prompted to inform the initial side temperatures for the grid. After that, when selecting Option 3, the program asks you to stimulate the grid at a specific point (you choose the point and value of temperature). Option 4 will ask you to inform the tolerance of the thermal distribution, calculate the thermal distribution based on the specific stimulation and initial conditions, and save the distribution on an image file. Option 5 will print the grid on the terminal. Lastly, the Option 6 will exit the program.