

## Thermal Distribution Simulation

Aman Sunesh (as18181)

### STEP 1: Problem Identification and Statement

The objective of this assignment is to develop a software to simulate thermal dissipation in a thin metal plate with constant (isothermal) temperatures on each side given a stimulation temperature and a point on the plate and form a thermal distribution grid representing the thermal data after simulation. Then, the software should store the resulting thermal distribution data in an image file.

### STEP 2: Gathering of Information and Input and Output Description

The temperature distribution in a thin metal plate with constant (isothermal) temperatures on each side can be modelled using a 2D grid, as shown in Figure 1

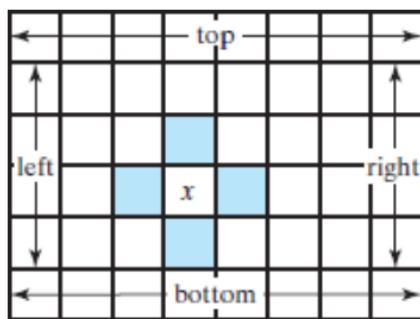


Figure 1: Thermal image distribution simulation.

As shown in the grid above, the metal plate is divided into discrete points, each with a coordinate  $(x, y)$ , where  $x$  represents the row and  $y$  represents the column. The temperature of each border (top, bottom, left, and right side) are set as some constant value (They may be the same or different). The temperature of an interior point can be computed as the average of the four adjacent temperatures; the points shaded in Figure 1 represent the adjacent temperatures for the point labelled  $x$  in the grid.

$$T_{x,y} = \frac{1}{4} (T_{x-1,y} + T_{x+1,y} + T_{x,y-1} + T_{x,y+1})$$

Each time that the temperature of an interior point changes, the temperatures of the points adjacent to it change. These changes continue until a thermal equilibrium is achieved and all temperatures become constant.

**Input Description:**

The data relevant to this problem consists of character values, unsigned integer values, integer values, Boolean values, and double values.

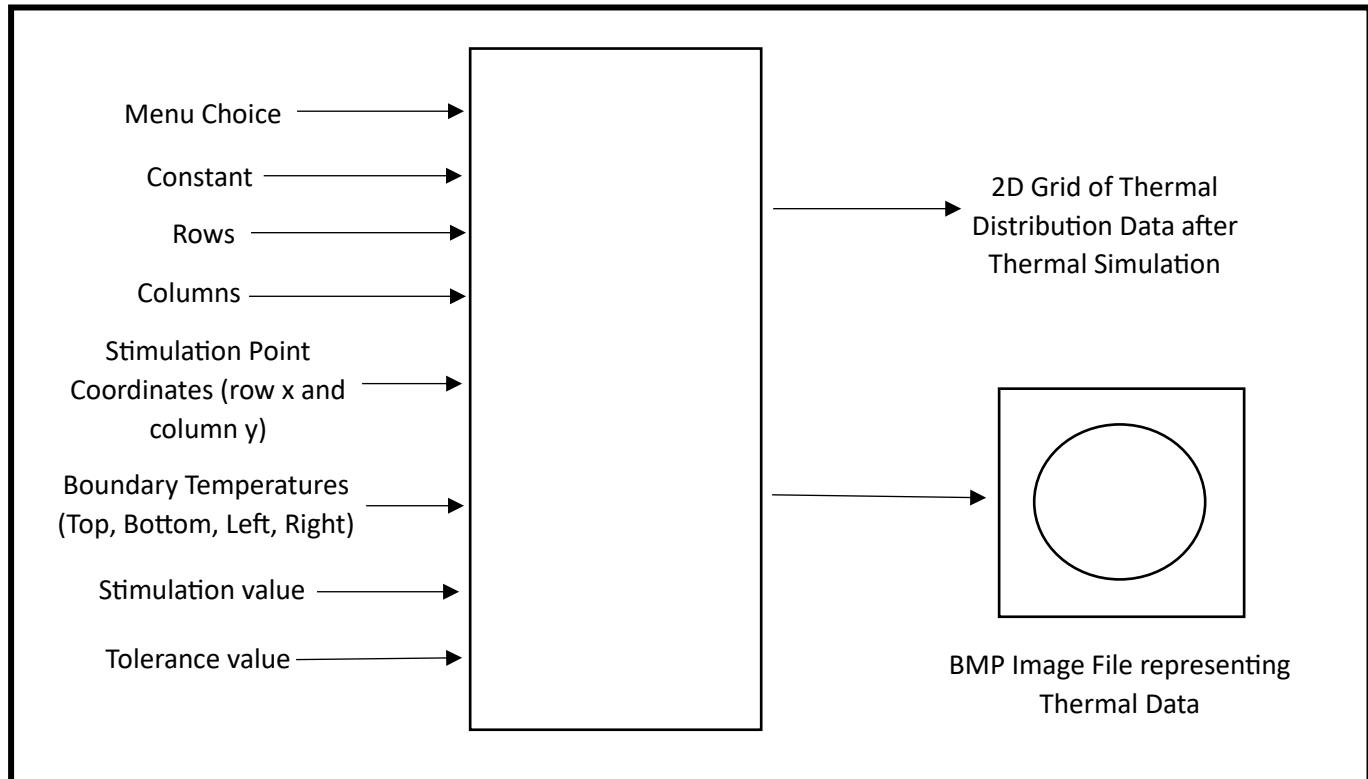
Beginning with choice (character value), which plays a vital role in navigating the program's menu.

- 1) If User Chooses 1) (Create the Dynamic 2D Grid) – create2DGrid(ROWS,COLS) is called:  
**Inputs:** ROWS, COLS
- 2) If User Chooses 2) (Set Boundary Conditions for the Grid) – setBoundaryConditions(arr, ROWS, COLS) is called:  
**Inputs:** Boundary Temperatures (Top, Bottom, Left, Right)
- 3) If User chooses 3) (Stimulate the Grid at a Specific Point) – stimulateGrid(arr, ROWS, COLS, x, y, constant) is called:  
**Inputs:** x-coordinate & y-coordinate of point of stimulation, Stimulation Value, constant
- 4) If User chooses 4) (Calculate the Thermal Distribution) – simulateThermalDissipation(arr, ROWS, COLS, x, y, constant) is called:  
**Inputs:** tolerance

**Output Description:**

1. 2D Grid of Thermal Distribution Data after Thermal Simulation
2. BMP Image File representing the Thermal Data

The I/O diagram for this problem is illustrated below:



**Figure 0.1: I/O Diagram**

**Description of Menu and Output Messages:**

**1. Menu Options:**

- Option 1: Create the Dynamic 2D Grid
- Option 2: Set Boundary Conditions for the Grid
- Option 3: Stimulate the Grid at a Specific Point
- Option 4: Calculate the Thermal Distribution
- Option 5: Print the Thermal Grid
- Option 6: Exit

## **2. User Interaction and Input Validation:**

- The program begins by displaying the menu options. The user is prompted to enter their choice (1, 2, 3, 4, 5, or 6).
- If the user enters a choice other than 1, 2, 3, 4, 5, or 6, the program displays an appropriate error message and prompts the user to re-enter a valid choice.
- If the user selects option 2 before selected option 1, then the program informs the user to select option 1 first as the program needs to create the dynamic 2D grid before setting the boundary conditions for the grid.
- Similarly, options 1 and 2 have to be selected before option 3. Options 1, 2, and 3 have to be selected before option 4. Option 1 has to be selected before option 5 as we must create a 2D grid before printing its contents.

## **3. Option 1: Create the Dynamic 2D Grid**

- The program prompts the user to enter the number of rows and columns for the grid.
- If the grid has not been created before, a new grid is created. Otherwise, the previously created grid is deallocated before a new grid is created.
- Input validation is implemented to ensure that the entered values are positive integers greater than 3.
- Memory is dynamically allocated for the 2D grid based on the user's input.

## **4. Option 2: Set Boundary Conditions for the Grid**

- The program prompts the user to enter initial temperatures for the top, bottom, left, and right sides of the grid.
- Input validation is applied to ensure that the entered values are positive and within the valid temperature range (0 to 255).
- Boundary temperatures are set in the grid based on user input, and all non-boundary elements are initialized to zero.

## **5. Option 3: Stimulate the Grid at a Specific Point**

- The program prompts the user to enter the coordinates (x and y) and stimulation value for a specific point in the grid.
- Input validation ensures that the coordinates are within the valid range and that the stimulation value is a positive number within the valid range (0 to 255).
- The specified point in the grid is stimulated with the entered value.
- The user is asked to select whether the stimulation should be constant or not.

## **6. Option 4: Calculate the Thermal Distribution**

- The program prompts the user to enter a tolerance value for equilibrium.
- Input validation ensures that the entered tolerance is a positive number within the valid range (0 to 255).

- The thermal dissipation simulation loop iterates until equilibrium is reached, and the resulting thermal distribution is calculated.
- The calculated thermal distribution is stored in a bitmap (BMP) image file.

## 7. Option 5: Print the Thermal Grid

- Displays the current state of the thermal grid.
- Checks if the grid has been created before attempting to print. Displays an error message if not.

## 8. Option 6: Exit

- Displays an exit message and terminates the program.
- After exiting from the switch structure, memory cleanup is done before exiting the program in main.

## STEP 3: Test Cases and Algorithm Design

### Test Cases:

#### 1) Menu Options

*Press 1 to create the Dynamic 2D Grid  
 Press 2 to Set Boundary Conditions for the Grid  
 Press 3 to Stimulate the Grid at a Specific Point  
 Press 4 to Calculate the Thermal Distribution  
 Press 5 to Print the Thermal Grid  
 Press 6 to quit the program*

*Please enter your choice:*

##### 1.1) User Selects Option 1

**Output:** *Please enter your choice: 1*

*Enter the number of rows of the grid:*

##### 1.2) User Selects Option 2

**Output:** *Please enter your choice: 2*

*Enter the initial temperature of the top side:*

##### 1.3) User Selects Option 3

**Output:** *Please enter your choice: 3*

*Enter the x-coordinate (columns - left to right) point of stimulation :*

**1.4)** User Selects Option 4

**Output:** Please enter your choice: 4  
Enter the tolerance value:

**1.5)** User Selects Option 5

**Output:** Please enter your choice: 5  
(Prints the thermal grid)

**1.6)** User Selects Option 6

**Output:** Please enter your choice: 6  
End of the program!

**1.7)** User Enters Invalid Input Example '@' for choice

**Output:** Please enter your choice: @  
Invalid Choice! Please enter either 1, 2, 3, 4, 5, or 6:

**1.8)** User Enters Value Outside the Permissible Range (1-6) for choice

**Output:** Please enter your choice: 165  
Invalid Choice! Please enter either 1, 2, 3, 4, 5, or 6:

## 2) Create the Dynamic 2D Grid

*Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

### 2.1) User Enters Valid Input (Array is Created)

**Output:** Enter the number of rows of the grid: 15  
Enter the number of columns of the grid: 15

### 2.2) User Enters -2 for Number of Rows

**Output:** Enter the number of rows of the grid: -2  
Please enter a positive integer value, greater than 3, for number of rows: 5  
Enter the number of columns of the grid: 6

### **2.3) User Enters 2 for Number of Rows and Columns**

**Output:** Enter the number of rows of the grid: 2

Please enter a positive integer value, greater than 3, for number of rows: 5

Enter the number of columns of the grid: 2

Please enter a positive integer value, greater than 3, for number of columns: 6

### **2.4) User Enters ‘apple’ as Input for Columns**

**Output:** Enter the number of rows of the grid: 50

Enter the number of columns of the grid: apple

Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive integer value, greater than 3, for number of columns: 4

### **2.5) User Enters ‘-‘ as Input for Rows**

**Output:** Enter the number of rows of the grid: -

Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive integer value, greater than 3, for number of rows: 4

Enter the number of columns of the grid: 4

## **3) Set Boundary Conditions for the Grid**

Press 1 to create the Dynamic 2D Grid

Press 2 to Set Boundary Conditions for the Grid

Press 3 to Stimulate the Grid at a Specific Point

Press 4 to Calculate the Thermal Distribution

Press 5 to Print the Thermal Grid

Press 6 to quit the program

Please enter your choice: 1

Enter the number of rows of the grid: 6

Enter the number of columns of the grid: 5

Please enter your choice: 2

### **3.1) User Enters Valid Input**

**Output:** Enter the initial temperature of the top side: 10

Enter the initial temperature of the bottom side: 20

Enter the initial temperature of the left side: 30

Enter the initial temperature of the right side: 40

### **3.2) Users Enters ‘&’ for initial temperature of the top side**

**Output:** Enter the initial temperature of the top side: @

Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side:

### **3.3) Users Enters ‘-5’ for initial temperature of the top side**

**Output:** Enter the initial temperature of the top side: @

Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side: -5

Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side: 6

### **3.4) User Enters 256 for initial temperature of the left side**

**Output:** Enter the initial temperature of the top side: 169

Enter the initial temperature of the bottom side: 48

Enter the initial temperature of the left side: 256

Please enter a positive value, between 0 and 255 (inclusive), for temperature of the left side:

### **3.5) User Prints the Data**

**Output:** Press 1 to create the Dynamic 2D Grid

Press 2 to Set Boundary Conditions for the Grid

Press 3 to Stimulate the Grid at a Specific Point

Press 4 to Calculate the Thermal Distribution

Press 5 to Print the Thermal Grid

Press 6 to quit the program

Please enter your choice: 1

Enter the number of rows of the grid: 6

Enter the number of columns of the grid: 5

Please enter your choice: 2

Enter the initial temperature of the top side: 10

Enter the initial temperature of the bottom side: 20

Enter the initial temperature of the left side: 30

Enter the initial temperature of the right side: 40

*Please enter your choice: 5*

10.00	10.00	10.00	10.00	10.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
20.00	20.00	20.00	20.00	20.00

#### **4) Stimulate the Grid at a Specific Point**

*Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 6*

*Enter the number of columns of the grid: 5*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 10*

*Enter the initial temperature of the bottom side: 20*

*Enter the initial temperature of the left side: 30*

*Enter the initial temperature of the right side: 40*

*Please enter your choice: 3*

##### **4.1) User Enters Valid Input**

**Output:** Please note that 0 represents the first column and 0 represents the first row.

*Enter the x-coordinate (columns-left to right) point of stimulation: 3*

*Enter the y-coordinate (rows -top to bottom) point of stimulation: 3*

*Enter the stimulation value: 255*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1*

#### **4.2) User Enters -5 for x-coordinate**

**Output:** Please note that 0 represents the first column and 0 represents the first row.

*Enter the x-coordinate (columns - left to right) point of stimulation : -5  
Please enter a value between 1 and (number of columns - 2), both inclusive,  
for the x-coordinate:*

#### **4.3) Users Enters a Value for y-coordinate that is Out of Range**

**Output:** Please note that 0 represents the first column and 0 represents the first row.

*Enter the x-coordinate (columns - left to right) point of stimulation : -5  
Please enter a value between 1 and (number of columns - 2), both inclusive,  
for the x-coordinate: 3*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 7  
Please enter a value between 1 and (number of rows - 2), both inclusive,  
or the y-coordinate: 2*

*Enter the stimulation value:*

#### **4.4) User Enters '@' for Stimulation Value**

**Output:** Please note that 0 represents the first column and 0 represents the first row.

*Enter the x-coordinate (columns - left to right) point of stimulation : 3*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 3*

*Enter the stimulation value: @*

*Error! Invalid Input! Please enter a valid numeric input!*

*Please enter a positive value, between 0 and 255 (inclusive), for the stimulation value:*

#### **4.5) User Enter Invalid Choice for the YES/NO Option**

**Output:** Please note that 0 represents the first column and 0 represents the first row.

*Enter the x-coordinate (columns - left to right) point of stimulation : 3*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 3*

*Enter the stimulation value: 100*

*Do you wish to keep the stimulation temperature constant?, Please enter 1  
for YES or 0 for NO: 4*

*Please enter either 0 or 1:*

#### **4.6) User Prints the Data**

**Output:** Press 1 to create the Dynamic 2D Grid

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 6*

*Enter the number of columns of the grid: 5*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 10*

*Enter the initial temperature of the bottom side: 20*

*Enter the initial temperature of the left side: 30*

*Enter the initial temperature of the right side: 40*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 3*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 3*

*Enter the stimulation value: 100*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 4*

*Please enter either 0 or 1: 1*

*Please enter your choice: 5*

10.00	10.00	10.00	10.00	10.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	100.00	40.00
30.00	0.00	0.00	0.00	40.00
20.00	20.00	20.00	20.00	20.00

## **5) Calculate the Thermal Distribution**

*Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 6*

*Enter the number of columns of the grid: 5*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 10  
Enter the initial temperature of the bottom side: 20  
Enter the initial temperature of the left side: 30  
Enter the initial temperature of the right side: 40*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 3  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 3  
Enter the stimulation value: 100*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1*

*Please enter your choice: 4*

### **5.1) User Enters Valid Input for Tolerance**

**Output:** *Enter the tolerance value: 1*

### **5.2) User Enters ‘!’ for Tolerance**

**Output:** *Enter the tolerance value: !*

*Error! Invalid Input! Please enter a valid numeric input!*

*Please enter a positive value, between 0 and 255 (inclusive), for the tolerance value:*

### **5.3) User Prints the Data (Stimulation is Constant)**

**Output:** *Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 6*

*Enter the number of columns of the grid: 5*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 10*

*Enter the initial temperature of the bottom side: 20*

*Enter the initial temperature of the left side: 30*

*Enter the initial temperature of the right side: 40*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 3*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 3*

*Enter the stimulation value: 100*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 4*

*Please enter either 0 or 1: 1*

*Please enter your choice: 4*

*Enter the tolerance value: 1*

*Please enter your choice: 5*

10.00	10.00	10.00	10.00	10.00
30.00	24.00	26.43	31.80	40.00
30.00	32.37	41.02	53.20	40.00
30.00	35.57	53.27	100.00	40.00
30.00	29.82	37.60	48.97	40.00
20.00	20.00	20.00	20.00	20.00

#### **5.4) User Prints the Data (Stimulation is Not Constant)**

*Output: Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 6*

*Enter the number of columns of the grid: 5*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 10*

*Enter the initial temperature of the bottom side: 20*

*Enter the initial temperature of the left side: 30*

*Enter the initial temperature of the right side: 40*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 3*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 3*

*Enter the stimulation value: 100*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 4*

*Please enter either 0 or 1: 0*

*Please enter your choice: 4*

*Enter the tolerance value: 1*

*Please enter your choice: 5*

10.00	10.00	10.00	10.00	10.00
30.00	20.34	19.60	24.92	40.00
30.00	24.41	24.37	29.91	40.00
30.00	25.82	26.17	30.33	40.00
30.00	24.15	24.18	27.88	40.00
20.00	20.00	20.00	20.00	20.00

## 6) Sample Cases to Obtain Thermal Distribution Image

### 6.1.1) When stimulation is constant

*Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 15*

*Enter the number of columns of the grid: 15*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 255*

*Enter the initial temperature of the bottom side: 100*

*Enter the initial temperature of the left side: 10*

*Enter the initial temperature of the right side: 50*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 8*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 8*

*Enter the stimulation value: 255*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1*

*Please enter your choice: 4*

*Enter the tolerance value: 1*

- The top side of the thermal image is expected to exhibit a prominent red shade, representing the highest initial temperature (255). Moving towards the center from the top, the red shade is anticipated to gradually fade away, transitioning into orange, yellow, and finally green due to thermal dissipation. The left side, with an initial temperature of 10, is expected to have a predominantly blue shade. The bottom side is expected to have a green shade due to its moderate temperature. The right side may exhibit a mix of shades, blending green and blue, owing to its moderately low initial temperature. The point of stimulation ( $x=8, y=8$ ) is expected to be represented by a circle with a red shade, reflecting the high stimulation temperature (255). The region surrounding this stimulated point is anticipated to exhibit a gradient, transitioning from red to yellow/orange, reflecting the dissipation effect.

#### **6.1.2) When stimulation is not constant**

*Press 1 to create the Dynamic 2D Grid*

*Press 2 to Set Boundary Conditions for the Grid*

*Press 3 to Stimulate the Grid at a Specific Point*

*Press 4 to Calculate the Thermal Distribution*

*Press 5 to Print the Thermal Grid*

*Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 15*

*Enter the number of columns of the grid: 15*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 255  
Enter the initial temperature of the bottom side: 100  
Enter the initial temperature of the left side: 10  
Enter the initial temperature of the right side: 50*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 8  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 8  
Enter the stimulation value: 255*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0*

*Please enter your choice: 4*

*Enter the tolerance value: 1*

- Similar to the previous case, the top side of the thermal image is expected to exhibit a prominent red shade, representing the highest initial temperature (255). Moving towards the center from the top, the red shade is anticipated to gradually fade away, transitioning into orange, yellow, and finally green due to thermal dissipation. The left side, with an initial temperature of 10, is expected to have a predominantly blue shade. The bottom side is expected to have a green shade due to its moderate temperature. The right side may exhibit a mix of shades, blending green and blue, owing to its moderately low initial temperature. In this case, the simulation is not constant, allowing observation of the shades of blue from the left and right sides moving towards the center due to thermal dissipation and the absence of constant stimulation. The lack of constant stimulation would enable the dynamic movement of temperature gradients Due to the high temperature of the top side and the moderate temperature of the bottom side, a green shade is expected at the center of the thermal image. The center serves as a convergence point for dissipated temperatures,

### **6.2.1) When stimulation is constant**

*Press 1 to create the Dynamic 2D Grid  
Press 2 to Set Boundary Conditions for the Grid  
Press 3 to Stimulate the Grid at a Specific Point  
Press 4 to Calculate the Thermal Distribution  
Press 5 to Print the Thermal Grid  
Press 6 to quit the program*

*Please enter your choice: 1*

*Enter the number of rows of the grid: 25*

*Enter the number of columns of the grid: 40*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 200*

*Enter the initial temperature of the bottom side: 200*

*Enter the initial temperature of the left side: 200*

*Enter the initial temperature of the right side: 200*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 10*

*Enter the y-coordinate (rows - top to bottom) point of stimulation : 10*

*Enter the stimulation value: 150*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1*

*Please enter your choice: 4*

*Enter the tolerance value: 1.25*

- Since the temperature of all the border sides are kept constant at 200 (high temperature), we can expect the border of the image having a red shade. Since the border elements have a constant temperature, we expect to observe heat dissipating uniformly as we approach the centre. As we move towards the centre from the border, we expect to see a orange ring followed by a yellow ring. The transition from red to orange and then yellow indicates a gradual decrease in temperature. Further towards the center, we expect to observe a green region, indicating a moderate temperature. The central area is characterized by a blue region, suggesting the lowest temperature within the green region. Now, we have constant stimulation in this case, but the stimulation value is not very high (150). So, at the point where x and y coordinates are 10, we expect to see a yellowish-orange circle depicting the moderately high temperature, and this stimulation would slight affect the neighbouring region. We expect to see a slight temperature elevation in the surrounding area due to constant stimulation.

### **6.2.2) When stimulation is not constant**

*Press 1 to create the Dynamic 2D Grid  
Press 2 to Set Boundary Conditions for the Grid  
Press 3 to Stimulate the Grid at a Specific Point  
Press 4 to Calculate the Thermal Distribution  
Press 5 to Print the Thermal Grid  
Press 6 to quit the program*

*Please enter your choice: 1  
Enter the number of rows of the grid: 25  
Enter the number of columns of the grid: 40*

*Please enter your choice: 2*

*Enter the initial temperature of the top side: 200  
Enter the initial temperature of the bottom side: 200  
Enter the initial temperature of the left side: 200  
Enter the initial temperature of the right side: 200*

*Please enter your choice: 3*

*Please note that 0 represents the first column and 0 represents the first row.*

*Enter the x-coordinate (columns - left to right) point of stimulation : 10  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 10  
Enter the stimulation value: 150*

*Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0*

*Please enter your choice: 4  
Enter the tolerance value: 1.25*

- Since the temperature of all the border sides are kept constant at 200 (high temperature), we can expect the border of the image having a red shade. Since the border elements have a constant temperature, we expect to observe heat dissipating uniformly as we approach the centre. As we move towards the centre from the border, we expect to see a orange ring followed by a yellow ring. The transition from red to orange and then yellow indicates a gradual decrease in temperature. Further towards the center, we expect to observe a green region, indicating a moderate temperature. The central area is characterized by a blue region, suggesting the lowest temperature within the green region. Since there is no constant stimulation in this case, there will be no disturbance in the region near the point with x and y coordinates as 10. The absence of constant stimulation is expected to allow a smooth and undisturbed transition of colors from red to blue in the central area.

### **6.3.1) When stimulation is constant**

Press 1 to create the Dynamic 2D Grid  
Press 2 to Set Boundary Conditions for the Grid  
Press 3 to Stimulate the Grid at a Specific Point  
Press 4 to Calculate the Thermal Distribution  
Press 5 to Print the Thermal Grid  
Press 6 to quit the program

Please enter your choice: 1  
Enter the number of rows of the grid: 15  
Enter the number of columns of the grid: 15

Please enter your choice: 2

Enter the initial temperature of the top side: 100  
Enter the initial temperature of the bottom side: 100  
Enter the initial temperature of the left side: 200  
Enter the initial temperature of the right side: 200

Please enter your choice: 3

Please note that 0 represents the first column and 0 represents the first row.

Enter the x-coordinate (columns - left to right) point of stimulation : 7  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 7  
Enter the stimulation value: 100

Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1

Please enter your choice: 4  
Enter the tolerance value: 1

- In this case, the left and right sides are expected to have a shade of red, representing the very high temperature (200) on these sides. The consistent red shade along the left and right borders is anticipated due to the maintenance of a constant high temperature. The top and bottom sides are expected to have a shade of blue, reflecting the comparatively low temperature on these sides. The blue color from the top and bottom sides is anticipated to merge towards the center from the borders. The top and bottom sides are expected to contribute to the central area's blue color, indicating the dissipation effect. The colors are expected to change from red to blue as they move from the left and right borders towards the center. The anticipated sequence includes a transition from red to orange, to yellow, to green, and finally to blue. Lastly, although constant stimulation is present in this case, the stimulation value is low (blue). A tiny blue circle is expected at the center (near the point where x and y coordinates are 7), indicating the low-temperature stimulation. The low stimulation value is not expected to significantly impact the neighbouring region.

### **6.3.2) When stimulation is not constant**

Press 1 to create the Dynamic 2D Grid

Press 2 to Set Boundary Conditions for the Grid

Press 3 to Stimulate the Grid at a Specific Point

Press 4 to Calculate the Thermal Distribution

Press 5 to Print the Thermal Grid

Press 6 to quit the program

Please enter your choice: 1

Enter the number of rows of the grid: 15

Enter the number of columns of the grid: 15

Please enter your choice: 2

Enter the initial temperature of the top side: 100

Enter the initial temperature of the bottom side: 100

Enter the initial temperature of the left side: 200

Enter the initial temperature of the right side: 200

Please enter your choice: 3

Please note that 0 represents the first column and 0 represents the first row.

Enter the x-coordinate (columns - left to right) point of stimulation : 7

Enter the y-coordinate (rows - top to bottom) point of stimulation : 7

Enter the stimulation value: 100

Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0

Please enter your choice: 4

Enter the tolerance value: 1

- In this case, the left and right sides are expected to have a shade of red, representing the very high temperature (200) on these sides. The consistent red shade along the left and right borders is anticipated due to the maintenance of a constant high temperature. The top and bottom sides are expected to have a shade of blue, reflecting the comparatively low temperature on these sides. The blue color from the top and bottom sides is anticipated to merge towards the center from the borders. The top and bottom sides are expected to contribute to the central area's blue color, indicating the dissipation effect. The colors are expected to change from red to blue as they move from the left and right borders towards the center. The anticipated sequence includes a transition from red to orange, to yellow, to green, and finally to blue. In this case, there is no constant stimulation, so the tiny blue circle present at the center in the previous case is not expected to be present. Other major changes due to the lack of constant stimulation are not anticipated as the stimulation value is low.

#### **6.4.1) When stimulation is not constant**

Press 1 to create the Dynamic 2D Grid

Press 2 to Set Boundary Conditions for the Grid

Press 3 to Stimulate the Grid at a Specific Point

Press 4 to Calculate the Thermal Distribution

Press 5 to Print the Thermal Grid

Press 6 to quit the program

Please enter your choice: 1

Enter the number of rows of the grid: 60

Enter the number of columns of the grid: 80

Please enter your choice: 2

Enter the initial temperature of the top side: 40

Enter the initial temperature of the bottom side: 60

Enter the initial temperature of the left side: 30

Enter the initial temperature of the right side: 75

Please enter your choice: 3

Please note that 0 represents the first column and 0 represents the first row.

Enter the x-coordinate (columns - left to right) point of stimulation : 30

Enter the y-coordinate (rows - top to bottom) point of stimulation : 40

Enter the stimulation value: 255

Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1

Please enter your choice: 4

Enter the tolerance value: 0.5

- We expect to get a very different thermal image in this case as compared to the other images. Here the values of all the four border elements are low. The stimulation value is extremely high, so the only region we expect to see a red shade is at the stimulation point. The values of all four border elements being low are expected to result in a predominantly blue shade across the image. The top and left sides, with very low temperatures, are anticipated to appear in a blue shade, which would continue until the center. The bottom and right sides have comparatively higher temperatures, due to which they are green in shade. Majority of the picture would be blue in colour. The right and bottom borders are expected to transition from green to blue as we move away from the borders towards the center. Since the stimulation point is very high, you would see a very sharp red circle at the stimulation point (Point where x-coordinate is 60 and y coordinate is 80). This would affect the region surrounding the stimulation point, resulting in a very visible green region surrounding this red circle, resulting from the impact of the high stimulation temperature.

#### **6.4.2) When stimulation is not constant**

Press 1 to create the Dynamic 2D Grid  
Press 2 to Set Boundary Conditions for the Grid  
Press 3 to Stimulate the Grid at a Specific Point  
Press 4 to Calculate the Thermal Distribution  
Press 5 to Print the Thermal Grid  
Press 6 to quit the program

Please enter your choice: 1  
Enter the number of rows of the grid: 60  
Enter the number of columns of the grid: 80

Please enter your choice: 2

Enter the initial temperature of the top side: 40  
Enter the initial temperature of the bottom side: 60  
Enter the initial temperature of the left side: 30  
Enter the initial temperature of the right side: 75

Please enter your choice: 3

Please note that 0 represents the first column and 0 represents the first row.

Enter the x-coordinate (columns - left to right) point of stimulation : 30  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 40  
Enter the stimulation value: 255

Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0

Please enter your choice: 4  
Enter the tolerance value: 0.5

- The thermal image in this case would be slightly different from case 6.4.1, where the temperature was constant. In this case, there is no constant stimulation. So the region with the highest temperature would be the right border, and hence it would be represented by the red shade. The bottom side has a temperature slightly lower than the right side, so even the bottom side would have a small region of red along its border. Both the right and the left side would change its colour from red to orange to yellow, to green and finally to blue as we move towards the centre. The left and top side have moderately low temperature, so they would be represented by green. They change into blue as we move towards the centre. There is a small circle we can see at the bottom right side of the image. This is the point of stimulation.

### **6.5.1) When stimulation is not constant**

Press 1 to create the Dynamic 2D Grid  
Press 2 to Set Boundary Conditions for the Grid  
Press 3 to Stimulate the Grid at a Specific Point  
Press 4 to Calculate the Thermal Distribution  
Press 5 to Print the Thermal Grid  
Press 6 to quit the program

Please enter your choice: 1  
Enter the number of rows of the grid: 20  
Enter the number of columns of the grid: 15

Please enter your choice: 2

Enter the initial temperature of the top side: 150  
Enter the initial temperature of the bottom side: 75  
Enter the initial temperature of the left side: 200  
Enter the initial temperature of the right side: 40

Please enter your choice: 3

Please note that 0 represents the first column and 0 represents the first row.

Enter the x-coordinate (columns - left to right) point of stimulation : 10  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 7  
Enter the stimulation value: 185

Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1

Please enter your choice: 4  
Enter the tolerance value: 0.8

- The left side of the thermal image is expected to have a red shade, representing the maximum temperature (200). Moving from the left border to the center, the color is expected to change from red to orange, to yellow, and finally to green due to thermal dissipation. The right side, with the minimum temperature, is anticipated to be represented by a blue shade. The top side, having a moderately high temperature, is expected to be represented by a yellow shade. The bottom side, with a slightly high temperature but less than the left and top sides, is expected to have a dark shade of green. Due to thermal dissipation and the merging of colors, the net color in the middle is anticipated to be a light shade of green. There would be a red circle at the point of stimulation (point where x coordinate is 10 and y coordinate is 7). The point of stimulation is expected to slightly increase the temperature of its neighbouring region, resulting in a small ring of yellow around the red circle.

### **6.5.2) When stimulation is not constant**

Press 1 to create the Dynamic 2D Grid

Press 2 to Set Boundary Conditions for the Grid

Press 3 to Stimulate the Grid at a Specific Point

Press 4 to Calculate the Thermal Distribution

Press 5 to Print the Thermal Grid

Press 6 to quit the program

Please enter your choice: 1

Enter the number of rows of the grid: 20

Enter the number of columns of the grid: 15

Please enter your choice: 2

Enter the initial temperature of the top side: 150

Enter the initial temperature of the bottom side: 75

Enter the initial temperature of the left side: 200

Enter the initial temperature of the right side: 40

Please enter your choice: 3

Please note that 0 represents the first column and 0 represents the first row.

Enter the x-coordinate (columns - left to right) point of stimulation : 10

Enter the y-coordinate (rows - top to bottom) point of stimulation : 7

Enter the stimulation value: 185

Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0

Please enter your choice: 4

Enter the tolerance value: 0.8

- The left side of the thermal image is expected to have a red shade, representing the maximum temperature (200). Moving from the left border to the center, the color is expected to change from red to orange, to yellow, and finally to green due to thermal dissipation. The right side, with the minimum temperature, is anticipated to be represented by a blue shade. The top side, having a moderately high temperature, is expected to be represented by a yellow shade. The bottom side, with a slightly high temperature but less than the left and top sides, is expected to have a dark shade of green. Due to thermal dissipation and the merging of colors, the net color in the middle-left region is anticipated to be a light shade of green. The extension of the blue region from the right border to the center aligns with the absence of constant stimulation and the lower temperatures of the bottom and right sides.

## **Algorithm Design:**

Declare and Initialize `NUM_SIDES` = 4  
Declare and Initialize `NEW_RANGE` = 255

Main Function:

    Declare choice  
    Declare and Initialize `ROWS` = 0, `COLS` = 0, and `constant` = 0  
    Declare and Initialize `arr` as double pointer to `nullptr`  
    Declare `x` and `y`

    Declare and Initialize `gridCreated` to false  
    Declare and Initialize `boundarySet` to false  
    Declare and Initialize `grisStimulated` to false

    Repeat while choice is not equal to 6

        {  
        Print newline  
        Print "-----", newline  
        Print " THERMAL DISTRIBUTION SIMULATION ", newline  
        Print "-----", newline  
        Print "1. Create the Dynamic 2D Grid", newline  
        Print "2. Set Boundary Conditions for the Grid", newline  
        Print "3. Stimulate the Grid at a Specific Point", newline  
        Print "4. Calculate the Thermal Distribution", newline  
        Print "5. Print the Thermal Grid", newline  
        Print "6. Exit", newline

    Print "Enter your choice (1-6):"  
    Read value into choice

    Set precision for output values

    Switch based on choice:

        If choice is equal to '1'

            If `arr` is equal to `nullptr`

                Call the function `create2DGrid`, passing `ROWS`, `COLS`, and assign the returned value to `arr`

                Set `gridCreated` flag to true

                break

        else

            Call the function `deallocate2Grid` passing `arr` and `ROWS`

            Call the function `create2DGrid`, passing `ROWS`, `COLS`, and assign the returned value to `arr`

            break

        If choice is equal to '2'

            If `gridCreated` is false

                Print "Error: Create the grid first (Option 1)!", newline

                break

            Call the `setBoundaryConditions` function, passing `arr`, `ROWS`, `COLS`

            Set `boundarySet` flag to true

            break

*If choice is equal to '3'*

*If gridCreated or boundarySet is false*

*Print "Error: Create the grid and set boundaries first (Options 1 and 2)!", newline*

*break*

*Call the stimulateGrid function, passing arr, ROWS, COLS, x, y, and constant*

*Set gridStimulated flag to true*

*break*

*If choice is equal to '4'*

*If gridCreated is false or boundarySet is false or gridStimulated is false*

*Print "Error: Create the grid, set boundaries, and stimulate the grid first (Options 1, 2, and 3)!"*

*break*

*Call simulateThermalDissipation function, passing arr, ROWS, COLS, x, y, and constant*

*break*

*If choice is equal to '5'*

*If gridCreated is false*

*Print "Error: Create the grid first (Option 1)!"*

*break*

*Call displayGrid function, passing arr, ROWS, COLS*

*break*

*If choice is equal to '6'*

*Print "End of the program!", newline*

*break*

*default case*

*Print "End of the program!"*

*Clear Input Buffer*

*Ignore input until newline*

*Continue*

*Call the function deallocate2DGrid passing arr and ROWS*

*Exit with code 0*

Define Function `create2DGrid(ROWS, COLS)`, with parameters `ROWS` and `COLS`

    Declare and Initialize `arr` as double pointer to `nullptr`

    Print “Enter the number of rows of the grid: “

    Read value into `ROWS`

    Repeat while `isValidInput` or `ROWS` less than 4

        Print “Please enter a positive integer value, greater than 3, for number of rows: “

        Read value into `ROWS`

        Print “Enter the number of columns of the grid: “

        Read value into `COLS`

    Repeat while `isValidInput` or `COLS` less than 4

        Print “Please enter a positive integer value, greater than 3, for number of columns: “

        Read value into `COLS`

    Allocate memory for the array of pointers `arr`

    Declare and Initialize `i = 0`

    Repeat while `i` less than `ROWS`

        Allocate memory for each row `arr[i]`

        Increment `i`

    Return `arr`

Define function `displayGrid(arr, ROWS, COLS)` with parameters `arr`, `ROWS`, and `COLS`

    Declare and Initialize `i` to 0

    Repeat while `i` less than `ROWS`

        Declare and Initialize `j` to 0

        Repeat while `j` less than `COLS`

            Print `arr[i][j]` with formatting

            Increment `j`

        Print newline

        Increment `i`

*Define function setBoundaryConditions(arr, ROWS, COLS) with parameters arr, ROWS, and COLS*

*Declare sides as array of double with size NUM\_SIZES*

*Print "Enter the initial temperature of the top side:"*

*Read value into sides[0]*

*Repeat while isInvalidInput or sides[0] < 0 or sides[0] > 255*

*Print ""Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side: "*

*Read value into sides[0]*

*Print "Enter the initial temperature of the bottom side:"*

*Read value into sides[1]*

*Repeat while isInvalidInput or sides[1] < 0 or sides[1] > 255*

*Print "Please enter a positive value, between 0 and 255 (inclusive), for temperature of the bottom side:*

*"*

*Read value into sides[1]*

*Print "Enter the initial temperature of the left side:"*

*Read value into sides[2]*

*Repeat while isInvalidInput or sides[2] < 0 or sides[2] > 255*

*Print "Please enter a positive value, between 0 and 255 (inclusive), for temperature of the left side: "*

*Read value into sides[2]*

*Print "Enter the initial temperature of the right side:"*

*Read value into sides[3]*

*Repeat while isInvalidInput or sides[3] < 0 or sides[3] > 255*

*Print ""Please enter a positive value, between 0 and 255 (inclusive), for temperature of the right side: "*

*Read value into sides[3]*

*Declare and Initialize i to 0*

*Repeat while i less than ROWS*

*Declare and Initialize j to 0*

*Repeat while j less than COLS*

If  $i$  is equal to 0

Assign  $sides[0]$  to  $arr[i][j]$

Otherwise if  $i$  is equal to  $ROWS - 1$

Assign  $sides[1]$  to  $arr[i][j]$

Otherwise if  $j$  is equal to 0

Assign  $sides[2]$  to  $arr[i][j]$

Otherwise if  $j$  is equal to  $COLS - 1$

Assign  $sides[3]$  to  $arr[i][j]$

Otherwise

Assign 0 to  $arr[i][j]$

Increment  $j$

Increment  $i$

Declare function  $stimulateGrid(arr, ROWS, COLS, x, y, constant)$  with parameters  $arr, ROWS, COLS, x, y$ , and  $constant$

Declare  $stimulation\_val$

Print "Please note that 0 represents the first column and 0 represents the first row.", newline

Print "Enter the x-coordinate (columns – left to right) point of stimulation: "

Read value into  $x$

Repeat while  $isValidInput$  or  $x >= COLS - 1$  or  $x <= 0$

Print "Please enter a value between 1 and (number of columns - 2), both inclusive, for the x-coordinate: "

Read value into  $x$

Print "Enter the y-coordinate (rows – top to bottom) point of stimulation: "

Read value into  $y$

Repeat while  $isValidInput$  or  $y >= ROWS - 1$  or  $y <= 0$

Print "Please enter a value between 1 and (number of rows - 2), both inclusive, for the y-coordinate: "

Read value into  $y$

*Print "Enter the stimulation value:"*

*Read value into stimulation\_val*

*Repeat while isInvalidInput or stimulation value < 0 or stimulation value > 255*

*Print "Please enter a positive value, between 0 and 255 (inclusive), for the stimulation value:"*

*Read value into stimulation\_val*

*Print "Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: ", newline*

*Read value into constant*

*Repeat while isInvalidInput or constant < 0 or constant > 1*

*Print "Please enter either 0 or 1: "*

*Read value into constant*

*Assign stimulation\_val to arr[y][x]*

*Define function simulateThermalDissipation(arr, ROWS, COLS, x, y, constant), with parameters arr, ROWS, COLS, x, y, and constant*

*Declare and initialize tolerance to 0*

*Declare equilibrium*

*Print "Enter the tolerance value:"*

*Read value into tolerance*

*Repeat while isInvalidInput or tolerance < 0 or tolerance > 255*

*Print "Please enter a positive value, between 0 and 255 (inclusive), for the tolerance value:"*

*Read value into tolerance*

*Repeat while not in equilibrium*

*Assign true to equilibrium*

*Declare and Initialize i to 0*

*Repeat while i less than ROWS - 1*

*Declare and Initialize j to 0*

*Repeat while j less than COLS - 1*

*If constant is equal to 1 and i is equal to y and j is equal to x*

*Continue*

*Otherwise*

*Declare and Initialize oldTemp to arr[i][j]*

*Declare and Initialize newTemp to (arr[i - 1][j] + arr[i + 1][j] + arr[i][j - 1] + arr[i][j + 1]) / 4*

*If absolute difference of oldTemp and newTemp is greater than or equal to tolerance*

*Assign newTemp to arr[i][j]*

*Assign false to equilibrium*

*Increment j*

*Increment i*

*Call the function convertDoubleToInt, passing arr, ROWS, COLS, and store the returned array in imageData*

*Call the function writeBitmap, passing "thermal\_distribution.bmp", imageData, COLS, ROWS*

*Declare and Initialize i to 0*

*Repeat while i less than ROWS*

*Delete individual rows imageData[i]*

*Increment i*

*Delete the array of pointers imageData*

*Define function convertDoubleToInt(arr, ROWS, COLS) with parameters arr, ROWS, and COLS*

*Declare and Initialize imageData as unsigned integer pointer to nullptr*

*Allocate memory for the array of pointers imageData*

*Declare and Initialize i = 0*

*Repeat while i less than ROWS*

*Allocate memory for each row imageData[i]*

*Increment i*

*Declare and Initialize maxval to arr[0][0]*

*Declare and Initialize minval to arr[0][0]*

*Declare and Initialize i to 0*

*Repeat while i less than ROWS*

*Declare and Initialize j to 0*

*Repeat while j less than COLS*

*If arr[i][j] greater than maxval*

*Assign arr[i][j] to maxval*

*If arr[i][j] less than minval*

*Assign arr[i][j] to minval*

*Increment j*

*Increment i*

*Declare and Initialize range to maxval – minval*

*Declare old\_value*

*Declare new\_value*

*Declare and Initialize i to 0*

*Repeat while i less than ROWS*

*Declare and Initialize j to 0*

*Repeat while j less than COLS*

*Declare and Initialize old\_value to arr[i][j]*

*Declare and Initialize new\_value to ((old\_value - minval) \* NEW\_RANGE / range + 0.5)  
        rounded to nearest integer*

*Convert new\_value into unsigned integers and assign to imageData[i][j]*

*Return imageData*

*Define function isInvalidInput()*

*Define and initialize invalidInput to false*

*If input fails*

*Assign true to invalidInput*

*Print "Error! Invalid Input! Please enter a valid numeric input!", newline*

*Print newline*

*Clear input buffer*

*Ignore input until newline*

*Otherwise*

*Assign false to invalidInput*

*Return invalidInput*

*Declare function deallocate2DGrid(arr;ROWS) with parameters arr, and ROWS*

*Declare and Initialize i to 0*

*Repeat while i less than ROWS*

*Delete individual rows arr[i]*

*Increment i*

*Delete the array of pointers arr*

#### STEP 4: Implementation

```
/*
 * Name: Aman Sunesh, NetID: as18181
 * Date: November 19, 2023
 * Program: MechanicalEngineering.cpp
 * Description: This program simulates
 * thermal dissipation in a thin metal
 * plate with constant (isothermal)
 * temperatures on each side given a
 * simulation temperature and a point on
 * the plate. The simulation calculates
 * the thermal distribution across the
 * system and is then converted to a
 * visual representation in a bitmap
 * image file. The program ensures proper
 * memory cleanup after execution
 */
#include <iostream>
#include <iomanip>
#include <cmath>
#include "Bitmap_Helper.h"
const int NUM_SIDES = 4;
const int NEW_RANGE = 255;

using namespace std;

//Function Declarations
double** create2DGrid(int& ROWS, int& COLS);
void displayGrid(double** arr, int ROWS, int COLS);
void setBoundaryConditions(double** arr, int ROWS, int COLS);
void stimulateGrid(double** arr, int ROWS, int COLS, int& x, int& y, int& constant);
void simulateThermalDissipation(double** arr, int ROWS, int COLS, int x, int y, int
constant);
uint8_t** convertDoubleToInt(double** arr, int ROWS, int COLS);
bool isValidInput();
void deallocate2DGrid(double** arr, int ROWS);
```

```

int main()
{
    //Declare and Initialize Variables
    int choice;
    int ROWS = 0, COLS = 0, constant = 0;
    double** arr = nullptr; //Pointer to a 2D array
    int x, y;

    //Flags to track which steps have been completed
    bool gridCreated = false;
    bool boundarySet = false;
    bool gridStimulated = false;

    //Main Simulation Loop
    do
    {
        //User Interface
        cout << endl;
        cout << "-----" << endl;
        cout << "          THERMAL DISTRIBUTION SIMULATION      " << endl;
        cout << "-----" << endl;
        cout << "1. Create the Dynamic 2D Grid" << endl;
        cout << "2. Set Boundary Conditions for the Grid" << endl;
        cout << "3. Stimulate the Grid at a Specific Point" << endl;
        cout << "4. Calculate the Thermal Distribution" << endl;
        cout << "5. Print the Thermal Grid" << endl;
        cout << "6. Exit" << endl;
        cout << "Enter your choice (1-6): ";
        cin >> choice;

        //Set precision for output values
        cout << fixed << setprecision(2);

        switch (choice)
        {
            case 1:
            {
                //Option 1: Create the Dynamic 2D Grid

                //If grid has not been created before
                if (arr == nullptr)
                {
                    arr = create2DGrid(ROWS, COLS);
                    gridCreated = true;
                    break;
                }
                else
                {
                    deallocate2DGrid(arr, ROWS);
                    arr = create2DGrid(ROWS, COLS);
                    break;
                }
            }

            case 2:
            {
                //Option 2: Set the Boundary Conditions for the Grid

                //Check if the user has created the grid
                if (!gridCreated)
                {
                    cout << "Error: Create the grid first (Option 1)!" << endl;
                    break;
                }
            }
        }
    }
}

```

```

        setBoundaryConditions(arr, ROWS, COLS);
        boundarySet = true;
        break;
    }

    case 3:
    {
        //Option 3: Stimulate the Grid at a Specific Point

        //Check if the user has created the grid and set
        boundaries

        if (!gridCreated || !boundarySet)
        {
            cout << "Error: Create the grid and set boundaries
first (Options 1 and 2)!" << endl;
            break;
        }

        stimulateGrid(arr, ROWS, COLS, x, y, constant);
        gridStimulated = true;
        break;
    }

    case 4:
    {
        //Option 4: Calculate the Thermal Distribution

        //Check if the user has created the grid, set boundaries,
        and stimulated the grid
        if (!gridCreated || !boundarySet || !gridStimulated)
        {
            cout << "Error: Create the grid, set boundaries,
and stimulate the grid first (Options 1, 2, and
3)!" << endl;

            break;
        }

        simulateThermalDissipation(arr, ROWS, COLS, x, y,
constant);
        break;
    }

    case 5:
    {
        //Option 5: Display the Thermal Grid

        //Check if the user has created the grid
        if (!gridCreated)
        {
            cout << "Error: Create the grid first (Option 1)!"
            << endl;
            break;
        }

        displayGrid(arr, ROWS, COLS);
        break;
    }
}

```

```

        case 6:
    {
        //Option 6: Exit

        cout << "End of the program!" << endl;
        //Switch statement breaks, Memory is deallocated in main
        //before exiting the program

        break;
    }

    default:
    {
        cout << "Invalid Choice! Please enter either 1, 2, 3, 4,
5, or 6: ";

        //Clears the input buffer
        cin.clear();

        //To ensure that error message is printed only once
        cin.ignore(numeric_limits<streamsize>::max(), '\n');
        continue;
    }
}

} while (choice != 6);

// Cleanup Allocated Memory
deallocate2DGrid(arr, ROWS);

return 0;
}

//Function to create a dynamic 2D grid
double** create2DGrid(int& ROWS, int& COLS)
{
    double** arr = nullptr; //Pointer to a 2D array

    //Prompt for the number of rows and validate input
    cout << "Enter the number of rows of the grid: ";
    cin >> ROWS;

    while (isValidInput() || ROWS < 4)
    {
        cout << "Please enter a positive integer value, greater than 3, for
        number of rows: ";
        cin >> ROWS;
    }

    //Prompt for the number of columns and validate input
    cout << "Enter the number of columns of the grid: ";
    cin >> COLS;
}

```

```

while (isValidInput() || COLS < 4)
{
    cout << "Please enter a positive integer value, greater than 3, for
number of columns: ";
    cin >> COLS;
}

//Allocate memory for the grid
arr = new double* [ROWS];

for (int i = 0; i < ROWS; i++)
{
    arr[i] = new double[COLS];
}

return arr;
}

//Function to display the thermal grid
void displayGrid(double** arr, int ROWS, int COLS)
{
    for (int i = 0; i < ROWS; i++)
    {
        for (int j = 0; j < COLS; j++)
        {
            cout << left << setw(7) << arr[i][j] << " ";
        }

        cout << endl;
    }
}

//Function to set boundary conditions for the grid
void setBoundaryConditions(double** arr, int ROWS, int COLS)
{
    double sides[NUM_SIDES];

    //Input boundary temperatures
    cout << "Enter the initial temperature of the top side: ";
    cin >> sides[0];

    while (isValidInput() || sides[0] < 0 || sides[0]>255)
    {
        cout << "Please enter a positive value, between 0 and 255 (inclusive),
for temperature of the top side: ";
        cin >> sides[0];
    }

    cout << "Enter the initial temperature of the bottom side: ";
    cin >> sides[1];

    while (isValidInput() || sides[1] < 0 || sides[1]>255)
    {
        cout << "Please enter a positive value, between 0 and 255 (inclusive),
for temperature of the bottom side: ";
        cin >> sides[1];
    }
}

```

```

cout << "Enter the initial temperature of the left side: ";
cin >> sides[2];

while (isValidInput() || sides[2] < 0 || sides[2] > 255)
{
    cout << "Please enter a positive value, between 0 and 255 (inclusive),\nfor temperature of the left side: ";
    cin >> sides[2];
}

cout << "Enter the initial temperature of the right side: ";
cin >> sides[3];

while (isValidInput() || sides[3] < 0 || sides[3] > 255)
{
    cout << "Please enter a positive value, between 0 and 255 (inclusive),\nfor temperature of the right side: ";
    cin >> sides[3];
}

//Set boundary temperatures in the grid
for (int i = 0; i < ROWS; i++)
{
    for (int j = 0; j < COLS; j++)
    {
        if (i == 0)
        {
            arr[i][j] = sides[0];
        }

        else if (i == ROWS - 1)
        {
            arr[i][j] = sides[1];
        }

        else if (j == 0)
        {
            arr[i][j] = sides[2];
        }

        else if (j == COLS - 1)
        {
            arr[i][j] = sides[3];
        }

        else
        {
            arr[i][j] = 0;
        }
    }
}

```

```

//Function to stimulate the grid at a specific point
void stimulateGrid(double** arr, int ROWS, int COLS, int& x, int& y, int& constant)
{
    double stimulation_val;

    //Input stimulation coordinates
    cout << "Please note that 0 represents the first column and 0 represents
the first row." << endl;

    cout << "Enter the x-coordinate (columns - left to right) point of
stimulation: ";
    cin >> x;

    while (isValidInput() || x >= COLS - 1 || x <= 0)
    {
        cout << "Please enter a value between 1 and (number of columns - 2),
both inclusive, for the x-coordinate: ";
        cin >> x;
    }

    cout << "Enter the y-coordinate (rows - top to bottom) point of stimulation ";
    cin >> y;

    while (isValidInput() || y >= ROWS - 1 || y <= 0)
    {
        cout << "Please enter a value between 1 and (number of rows - 2), both
inclusive, for the y-coordinate: ";
        cin >> y;
    }

    //Input stimulation value
    cout << "Enter the stimulation value: ";
    cin >> stimulation_val;

    while (isValidInput() || stimulation_val < 0 || stimulation_val > 255)
    {
        cout << "Please enter a positive value, between 0 and 255 (inclusive),
for the stimulation value: ";
        cin >> stimulation_val;
    }

    //Check if user wants the stimulation to be constant or discrete
    cout << "Do you wish to keep the stimulation temperature constant?, Please
enter 1 for YES or 0 for NO: ";
    cin >> constant;

    while (isValidInput() || constant<0 || constant>1)
    {
        cout << "Please enter either 0 or 1: ";
        cin >> constant;
    }

    //Apply stimulation to the grid
    arr[y][x] = stimulation_val;
}

```

```

//Function to stimulate thermal dissipation in the grid
void simulateThermalDissipation(double** arr, int ROWS, int COLS, int x, int y, int
constant)
{
    double tolerance = 0;
    bool equilibrium;

    //Input tolerance for equilibrium
    cout << "Enter the tolerance value: ";
    cin >> tolerance;

    while (isValidInput() || tolerance < 0 || tolerance > 255)
    {
        cout << "Please enter a positive value, between 0 and 255 (inclusive),\nfor the tolerance value: ";
        cin >> tolerance;
    }

    //Thermal dissipation simulation loop
    do
    {
        equilibrium = true;

        //Iterate through the grid cells for thermal dissipation
        for (int i = 1; i < ROWS - 1; i++)
        {
            for (int j = 1; j < COLS - 1; j++)
            {
                //Continuous Stimulation
                if (constant == 1 && i == y && j == x)
                {
                    continue;
                }

                else
                {
                    double oldTemp = arr[i][j];
                    double newTemp = (arr[i - 1][j] + arr[i + 1][j] +
arr[i][j - 1] + arr[i][j + 1]) / 4;

                    //Check for equilibrium
                    if (abs(oldTemp - newTemp) >= tolerance)
                    {
                        arr[i][j] = newTemp;
                        equilibrium = false;
                    }
                }
            }
        }
    } while (!equilibrium);

    //Convert double data to unsigned integers
    uint8_t** imageData = convertDoubleToInt(arr, ROWS, COLS);

    //Store the thermal data in the form of an image in .bmp format
    writeBitmap("thermal_distribution.bmp", imageData, COLS, ROWS); //Assuming
    name of the image file is given
}

```

```

//Cleanup allocated memory for bitmap data
for (int i = 0; i < ROWS; ++i) {
    delete[] imageData[i];
}
delete[] imageData;

}

//Function to convert double data to unsigned integers
uint8_t** convertDoubleToInt(double** arr, int ROWS, int COLS)
{
    //Create a Uint_8 array
    uint8_t** imageData = nullptr;

    //Allocate memory for the new array
    imageData = new uint8_t * [ROWS];

    for (int i = 0; i < ROWS; i++)
    {
        imageData[i] = new uint8_t[COLS];
    }

    //Find the maximum and minimum values in the input array
    int maxval = arr[0][0];
    int minval = arr[0][0];

    //Iterate through the array to determine the range of values
    for (int i = 0; i < ROWS; i++)
    {
        for (int j = 0; j < COLS; j++)
        {
            //Update maxval if the current element is greater
            if (arr[i][j] > maxval)
            {
                maxval = arr[i][j];
            }

            //Update minval if the current element is smaller
            if (arr[i][j] < minval)
            {
                minval = arr[i][j];
            }
        }
    }

    //Calculate the range of values
    int range = maxval - minval;

    //Variables for value scaling and rounding
    int old_value;
    int new_value;

    /*
        New Range = (NewMax - NewMin);
        NewValue = (((Old Value) - OldMin) * NewRange) / OldRange) + NewMin;
    */
}

```

```

//Convert each element from the input array to uint8_t and store in the new
//array
for (int i = 0; i < ROWS; i++)
{
    for (int j = 0; j < COLS; j++)
    {
        //Get the old (original) value from the input array
        int old_value = arr[i][j];

        //Scale the old value to fit the new range (0-255) and round to
        //the nearest integer
        int new_value = static_cast<int>((old_value - minval) * NEW_RANGE
        / range + 0.5);

        //Store the scaled and rounded value as uint8_t in the new array
        imageData[i][j] = static_cast<uint8_t>(new_value);
    }
}

//Return the new 2D array of uint8_t
return imageData;
}

//Function to check if the input is valid
bool isValidInput()
{
    bool invalidInput = false;

    //Check for invalid input
    if (cin.fail())
    {
        invalidInput = true;
        cout << "Error! Invalid Input! Please enter a valid numeric input!" <<
        endl;
        cout << endl;

        //Clears the input buffer
        cin.clear();

        //To ensure that error message is printed only once
        cin.ignore(numeric_limits<streamsize>::max(), '\n');
    }

    else
    {
        invalidInput = false;
    }

    return invalidInput;
}

void deallocate2DGrid(double** arr, int ROWS)
{
    for (int i = 0; i < ROWS; ++i)
    {
        delete[] arr[i];
    }

    delete[] arr;
}

```

## **STEP 5: Test and Verification (and Debugging)**

### **Test Cases 1.1-1.6:**

The output for each of these test cases from the program is in agreement with the test case expected output. This can be seen from the Sample Output section that follows this section.

**Therefore, we can conclude that the program is functioning correctly.**

### **Test Case 1.7: If user selects '@' as choice**

*Invalid Choice! Please enter either 1, 2, 3, 4, 5, or 6:*

,which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

### **Test 1.8: If user selected 165 as choice**

*Invalid Choice! Please enter either 1, 2, 3, 4, 5, or 6:*

,which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

### **Test Cases 2.1-2.3:**

The output for each of these test cases from the program is in agreement with the test case expected output. This can be seen from the Sample Output section that follows this section.

**Therefore, we can conclude that the program is functioning correctly.**

### **Test Case 2.4: If user selects 'apple' as Input for columns**

*Error! Invalid Input! Please enter a valid numeric input!  
Please enter a positive integer value, greater than 3, for number of columns:*

,which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test 2.5: If user enters ‘-’ as input for rows**

*Error! Invalid Input! Please enter a valid numeric input!  
Please enter a positive integer value, greater than 3, for number of columns:*

,which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Cases 3.1-3.4:**

The output for each of these test cases from the program is in agreement with the test case expected output. This can be seen from the Sample Output section that follows this section.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 3.5: If the user prints the data, with input as given in option 3.5**

10.00	10.00	10.00	10.00	10.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
20.00	20.00	20.00	20.00	20.00

,which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 4.1-4.5:**

The output for each of these test cases from the program is in agreement with the test case expected output. This can be seen from the Sample Output section that follows this section.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 4.6: If the user prints the data, with input as given in option 3.5**

10.00	10.00	10.00	10.00	10.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	0.00	40.00
30.00	0.00	0.00	100.00	40.00
30.00	0.00	0.00	0.00	40.00
20.00	20.00	20.00	20.00	20.00

,which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 5.1-5.2:**

The output for each of these test cases from the program is in agreement with the test case expected output. This can be seen from the Sample Output section that follows this section.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 5.3: If the user prints the data, with input as given in option 5.3**

10.00	10.00	10.00	10.00	10.00
30.00	24.00	26.43	31.80	40.00
30.00	32.37	41.02	53.20	40.00
30.00	35.57	53.27	100.00	40.00
30.00	29.82	37.60	48.97	40.00
20.00	20.00	20.00	20.00	20.00

,which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 5.4: If the user prints the data, with input as given in option 3.5**

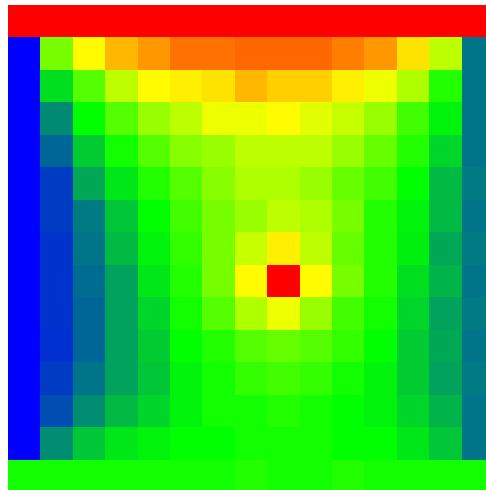
10.00	10.00	10.00	10.00	10.00
30.00	20.34	19.60	24.92	40.00
30.00	24.41	24.37	29.91	40.00
30.00	25.82	26.17	30.33	40.00
30.00	24.15	24.18	27.88	40.00
20.00	20.00	20.00	20.00	20.00

, which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 6.1.1:** Considering the data provided (as given in Step 3), the output (thermal grid) is

**The thermal image for this case is:**



## Analysis:

## **Top Side Gradient:**

- The top side demonstrates a visually apparent gradient from red to green, validating the expected thermal dissipation.

## **Left Side Blue Shade:**

- The left side prominently exhibits a blue shade, in accordance with the anticipated low temperature.

## Stimulated Point and Surroundings:

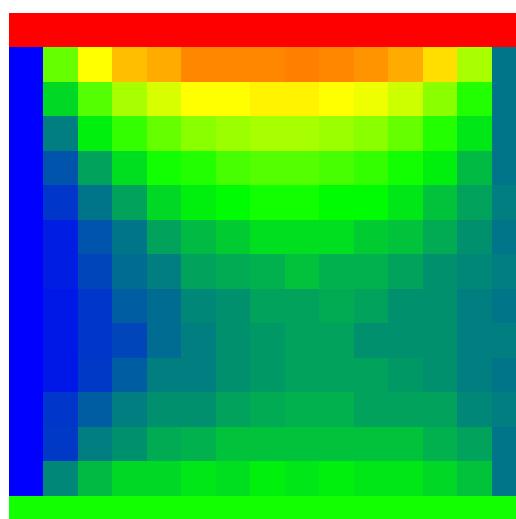
- The stimulated point ( $x=8$ ,  $y=8$ ) is represented by a red-shaded circle, aligning with the expected high stimulation temperature.
  - The neighbouring region displays a discernible gradient from red to yellow/orange, validating the dissipation effect.

, which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 6.1.2:** Considering the data provided (as given in Step 3), the output (thermal grid) is

**The thermal image for this case is:**



## **Analysis:**

### **Top Side Gradient:**

- The top side demonstrates the anticipated gradient from red to green, affirming the simulation's ability to simulate thermal dissipation.

### **Left Side Blue Shade:**

- The left side prominently exhibits a blue shade, aligning with the expected low temperature.

### **Bottom Side Green Shade:**

- The bottom side displays the anticipated green shade, in line with the expected moderate temperature.

### **Right Side Blending Shades:**

- The right side shows a mix of green and blue shades, validating the anticipated blending effect.

### **Dynamic Gradients:**

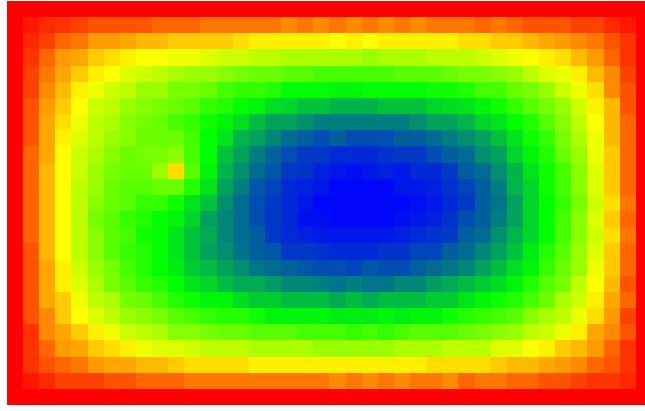
- The dynamic movement of blue shades from the left and right sides towards the centre aligns with the expected behaviour under variable stimulation.
- Green shades from both the top and bottom regions contribute to dynamic gradients towards the center.

, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.2.1: Considering the data provided (as given in Step 3), the output (thermal grid) is**

**The thermal image for this case is:**



## **Analysis:**

## Constant Temperature at Border (200):

- The entire border of the image exhibits a constant high temperature (200), resulting in a uniform red shade.
  - The consistent red shade along the border aligns with the expectation of a constant high temperature.

## Uniform Heat Dissipation Towards Center:

- As anticipated, there is a uniform dissipation of heat as we move from the border towards the center.
  - The transition from red to orange and then yellow signifies a gradual decrease in temperature towards the center.

**Color Transitions: Red → Orange Ring → Yellow Ring → Green Region:**

- The observed color transitions align with the expected behavior, confirming the simulation's ability to capture gradual temperature changes.
  - The sequence of color transitions, from red at the border to green in the central region, corresponds to the simulated dissipation process.

**Central Blue Region Within Green:**

- Further towards the center, the image exhibits a green region, indicating a moderate temperature.
- The central area is characterized by a blue region, suggesting the lowest temperature within the green region.
- The observed blue region in the center supports the simulation's accuracy in representing temperature variations.

**Constant Stimulation at (x=10, y=10 - Stimulation Value: 150):**

- At the stimulation point (x=10, y=10), a yellowish-orange circle is observed, depicting a moderately high temperature due to constant stimulation.
- The surrounding area shows a slight temperature elevation, confirming the localized impact of constant stimulation.

**Dynamic Gradients and Temperature Variation:**

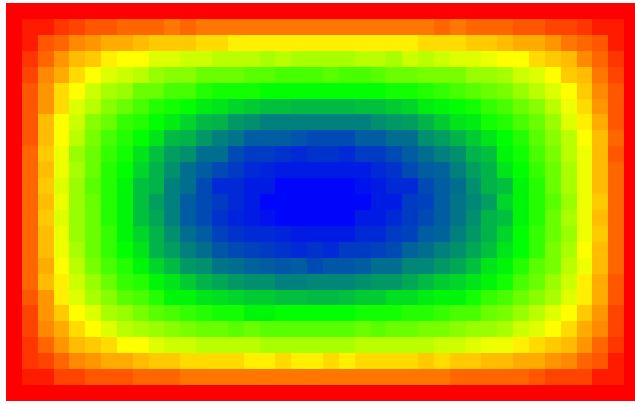
- The simulation successfully captures dynamic gradients, showcasing the movement from high temperatures at the border to lower temperatures at the center.
- The dynamic response to constant stimulation, especially the yellowish-orange circle and its influence on the neighbouring region, aligns with the expected behaviour.

, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.2.2: Considering the data provided (as given in Step 3), the output (thermal grid) is**

**The thermal image for this case is:**



## Analysis:

## Constant Temperature at Border (200):

- The entire border of the image exhibits a constant high temperature (200), resulting in a uniform red shade.
  - The consistent red shade along the border aligns with the expectation of a constant high temperature.

## Uniform Heat Dissipation Towards Center:

- As anticipated, there is a uniform dissipation of heat as we move from the border towards the center.
  - The transition from red to orange and then yellow signifies a gradual decrease in temperature towards the center.

**Color Transitions: Red → Orange Ring → Yellow Ring → Green Region:**

- The observed color transitions align with the expected behavior, confirming the simulation's ability to capture gradual temperature changes.
  - The sequence of color transitions, from red at the border to green in the central region, corresponds to the simulated dissipation process.

**Central Blue Region Within Green:**

- Further towards the center, the image exhibits a green region, indicating a moderate temperature.
- The central area is characterized by a blue region, suggesting the lowest temperature within the green region.
- The observed blue region in the center supports the simulation's accuracy in representing temperature variations.

**No Constant Stimulation ( $x=10, y=10$ ):**

- Since there is no constant stimulation in this case, there is no disturbance in the region near the point with  $x$  and  $y$  coordinates as 10.
- The absence of constant stimulation allows for an undisturbed transition of colors from red to blue in the central area.

**Dynamic Gradients and Temperature Variation:**

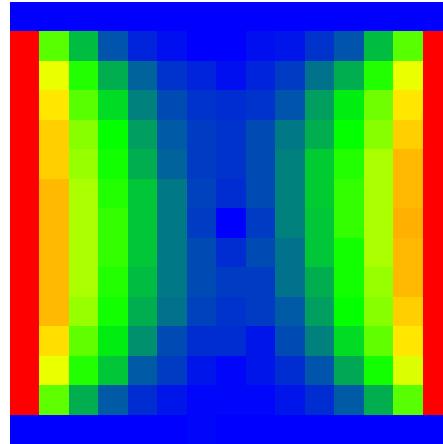
- The simulation successfully captures dynamic gradients, showcasing the movement from high temperatures at the border to lower temperatures at the center.
- The lack of constant stimulation contributes to a smooth and undisturbed transition of colors, emphasizing the natural dissipation process.

, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.3.1: Considering the data provided (as given in Step 3), the output (thermal grid) is**

**The thermal image for this case is:**



## Analysis:

### **Constant Temperature on Left and Right Sides (200):**

- The left and right sides indeed exhibit a consistent red shade, consistent with the expectation of a very high temperature (200) on these borders.

## **Top and Bottom Sides:**

- The top and bottom sides accurately display a shade of blue, aligning with the anticipated lower temperature on these borders.

#### **Contribution to Central Blue Color:**

- As expected, the blue color from the top and bottom sides seamlessly merges towards the center from the borders.
  - This contributes to the central area's blue color, reflecting the dissipation effect.

**Color Transition Sequence: Red → Orange → Yellow → Green → Blue:**

- The observed transition from red to orange, to yellow, to green and finally to blue towards the center corresponds well with the expected gradual decrease in temperature.

**Constant Stimulation with Low Value (Blue):**

- True to the prediction, a small blue circle is visible at the center (near the point where x and y coordinates are 7), indicating low-temperature stimulation.
- The low stimulation value is evident, and its limited impact on the neighboring region is observable.

**Dynamic Gradients and Temperature Variation:**

- The simulation successfully captures dynamic gradients, showcasing the movement from high temperatures at the border to lower temperatures at the center.
- The lack of significant disturbance in the region near the point with x and y coordinates as 7 due to low constant stimulation is evident.

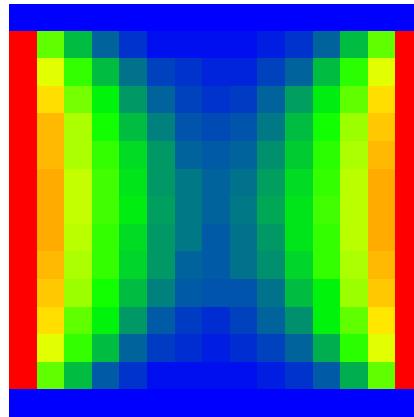
, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.3.2: Considering the data provided (as given in Step 3), the output (thermal grid) is**

100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
200.00	146.81	124.51	113.32	107.99	104.81	103.30	102.10	102.01	104.80	108.03	113.80	124.10	146.89	200.00				
200.00	163.98	140.19	124.36	114.98	109.19	106.29	104.32	105.05	109.18	114.81	124.33	139.44	163.47	200.00				
200.00	171.38	149.16	132.68	121.90	114.22	109.58	107.50	108.35	113.35	120.26	131.78	148.32	171.25	200.00				
200.00	175.16	154.74	137.93	125.73	117.39	112.68	110.32	111.36	116.90	124.96	136.51	153.32	174.41	200.00				
200.00	177.62	157.86	141.67	129.03	119.68	114.70	112.49	113.66	119.18	127.39	139.73	156.27	176.56	200.00				
200.00	178.66	159.67	142.99	130.05	120.94	115.49	113.77	115.22	121.23	129.56	141.52	157.89	177.87	200.00				
200.00	178.59	159.14	142.82	131.05	121.66	116.48	114.99	115.96	122.17	130.62	142.80	158.33	178.33	200.00				
200.00	177.73	157.93	141.85	129.66	120.73	115.42	113.75	115.33	120.89	130.40	141.96	157.91	178.27	200.00				
200.00	176.71	156.92	140.88	128.63	120.01	114.56	112.87	114.31	119.39	128.13	140.58	156.90	176.86	200.00				
200.00	174.59	153.43	137.24	125.43	117.44	112.67	111.15	112.82	116.72	124.69	136.56	153.57	174.68	200.00				
200.00	171.82	148.77	132.86	120.83	112.92	108.65	107.44	109.09	113.44	119.92	131.27	148.68	170.67	200.00				
200.00	163.92	140.53	124.62	114.46	108.56	105.44	104.79	106.35	109.19	114.10	123.76	139.21	163.20	200.00				
200.00	146.93	124.82	114.18	107.27	103.98	102.65	102.44	102.33	104.15	107.28	113.12	123.77	146.74	200.00				
100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

The thermal image for this case is:



### Analysis:

#### Constant Temperature on Borders:

- The observed consistent red shade along the left and right borders aligns with the expected very high temperature (200) on these sides.

#### Top and Bottom Sides:

- The blue shade on the top and bottom sides is consistent with the prediction, indicating the lower temperature on these borders.

#### Contribution to Central Blue Color:

- As expected, the blue color from the top and bottom sides seamlessly merges towards the center, contributing to the central area's blue color.

**Color Transition Sequence: Red → Orange → Yellow → Green → Blue:**

- The observed transition from red to blue as we move from the left and right borders towards the center corresponds well with the expected gradual decrease in temperature.

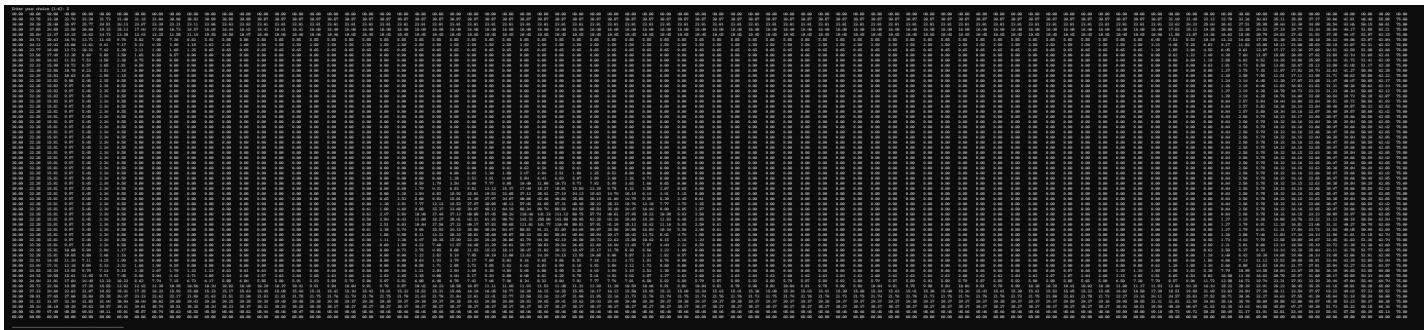
**No Constant Stimulation:**

- The absence of a constant stimulation is evident by the lack of the small blue circle at the center, aligning with the expectation in this case.

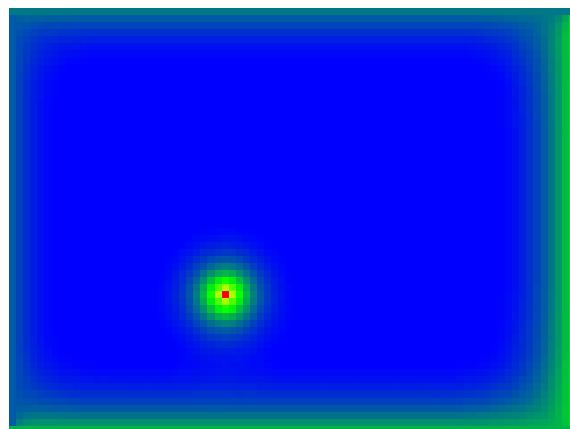
, which is in agreement with the test case expected output.

Therefore, we can conclude that the program is functioning correctly.

**Test Case 6.4.1: Considering the data provided (as given in Step 3), the output (thermal grid) is**



The thermal image for this case is:



## **Analysis:**

### **Predominantly Blue Image:**

- As anticipated, the majority of the image exhibits a blue shade due to the low temperatures on all four border elements.

### **Sharp Red Circle at Stimulation Point:**

- The presence of a very sharp red circle at the stimulation point aligns with the expectation of an extremely high stimulation value.

### **Blue Shade on Top and Left Sides:**

- The appearance of a blue shade on the top and left sides corresponds well with the prediction of low temperatures on these sides.

### **Green Shade on Bottom and Right Sides:**

- The observed green shade on the bottom and right sides is consistent with the expectation of comparatively higher temperatures on these borders.

### **Color Transition from Green to Blue:**

- The transition from green to blue as we move away from the right and bottom borders towards the center aligns with the anticipated color variation.

### **Visible Green Region Surrounding Stimulation Point:**

- The visible green region surrounding the stimulation point supports the expectation of a noticeable impact on the region due to the high stimulation temperature.

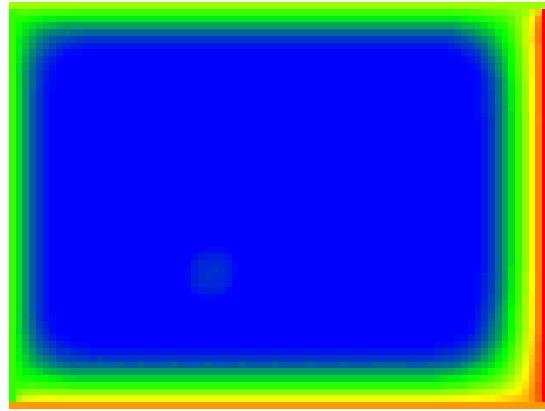
, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.4.2: Considering the data provided (as given in Step 3), the output (thermal grid) is**

```
1. Create the System of Eq's
2. Solve the System of Eq's
3. Calculate the Grid & Sensitive Node
4. Plot the Thermal Grid
```

**The thermal image for this case is:**



**Analysis:**

#### **Predominantly Blue Image:**

As anticipated, the majority of the image exhibits a blue shade due to the low temperatures on all four border elements.

#### **Sharp Red Circle at Stimulation Point:**

The presence of a very sharp red circle at the stimulation point aligns with the expectation of an extremely high stimulation value.

#### **Blue Shade on Top and Left Sides:**

The appearance of a blue shade on the top and left sides corresponds well with the prediction of low temperatures on these sides.

**Green Shade on Bottom and Right Sides:**

The observed green shade on the bottom and right sides is consistent with the expectation of comparatively higher temperatures on these borders.

**Color Transition from Green to Blue:**

The transition from green to blue as we move away from the right and bottom borders towards the center aligns with the anticipated color variation.

**Visible Green Region Surrounding Stimulation Point:**

- The visible green region surrounding the stimulation point supports the expectation of a noticeable impact on the region due to the high stimulation temperature.

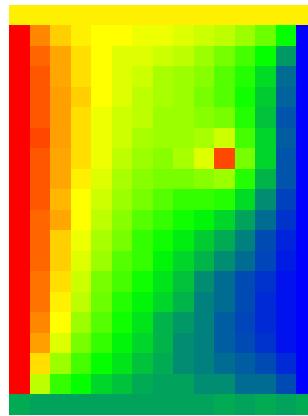
, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.5.1: Considering the data provided (as given in Step 3), the output (thermal grid) is**

150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00	150.00
200.00	172.20	159.38	152.44	148.57	145.99	144.16	142.14	139.98	137.20	132.86	126.91	116.41	94.67	40.00		
200.00	179.43	164.86	154.71	147.90	142.88	139.36	136.19	132.55	127.01	120.22	110.35	96.12	73.26	40.00		
200.00	181.67	166.90	155.64	147.41	141.05	136.13	131.69	127.04	121.04	113.48	101.25	85.31	64.44	40.00		
200.00	182.33	167.42	155.57	145.96	138.77	133.34	129.30	124.81	119.54	111.40	98.01	81.39	61.06	40.00		
200.00	183.19	167.81	155.23	145.10	137.60	132.10	128.36	125.34	123.09	117.35	100.92	81.19	60.39	40.00		
200.00	182.63	167.32	154.42	143.85	136.39	130.13	126.72	127.08	131.20	135.19	109.06	83.08	60.32	40.00		
200.00	181.93	166.36	153.43	142.54	133.98	127.27	124.34	127.06	140.66	185.00	119.74	84.90	60.58	40.00		
200.00	181.64	165.55	151.42	139.73	130.70	123.47	119.17	119.07	122.20	127.03	102.70	78.26	58.11	40.00		
200.00	180.97	163.79	148.91	136.49	126.45	117.96	111.77	107.86	105.11	100.90	86.57	70.24	54.63	40.00		
200.00	180.39	162.61	146.99	133.70	122.82	113.08	105.01	98.60	92.53	85.86	75.39	63.48	51.18	40.00		
200.00	178.98	160.26	143.76	129.45	118.05	107.46	98.68	90.94	83.57	76.56	67.54	58.05	48.54	40.00		
200.00	178.25	157.70	140.31	125.31	113.41	102.78	93.56	85.06	77.13	70.26	62.07	54.51	46.88	40.00		
200.00	176.34	154.95	136.51	120.87	108.32	97.65	88.53	80.62	72.73	65.29	58.80	51.97	45.42	40.00		
200.00	175.00	151.93	132.94	116.39	103.32	92.74	84.30	76.16	68.91	62.21	55.87	50.09	44.79	40.00		
200.00	171.73	147.60	127.91	111.25	98.69	88.74	80.76	73.84	67.47	60.78	55.25	49.69	44.58	40.00		
200.00	167.02	139.79	119.85	104.91	93.65	85.09	78.20	71.94	66.33	61.35	56.49	51.80	45.89	40.00		
200.00	156.57	127.51	108.74	96.09	87.16	80.84	76.08	71.44	67.34	63.71	59.79	55.13	49.19	40.00		
200.00	134.62	106.91	93.49	86.45	81.22	77.26	74.91	72.26	70.73	68.31	65.72	62.64	56.71	40.00		
75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00

The thermal image for this case is:



### Analysis:

#### Temperature Gradient on Left Side:

- The observed red shade on the left side aligns with the expectation of the maximum temperature on this side.

#### Color Transition from Red to Green:

- The observed transition from red to orange, to yellow, and finally to green as we move from the left border to the center corresponds to the anticipated thermal dissipation.

**Representation of Minimum Temperature on Right Side:**

- The appearance of a blue shade on the right side aligns with the expectation of the minimum temperature on this side.

**Yellow Shade on Top Side:**

- The presence of a yellow shade on the top side is consistent with the anticipation of a moderately high temperature.

**Dark Green Shade on Bottom Side:**

- The observed dark green shade on the bottom side corresponds to the expectation of a slightly high temperature but less than the left and top sides.

**Net Light Shade of Green in the Middle:**

- The overall light shade of green in the middle supports the prediction of color merging due to thermal dissipation.

**Representation of Stimulation Point:**

- The presence of a red circle at the point of stimulation aligns with the expected high temperature at that point.

**Increase in Temperature in Surrounding Region:**

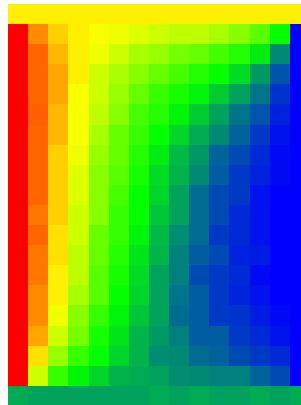
- The small ring of yellow around the red circle indicates the anticipated increase in temperature in the surrounding region due to the stimulation.

, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

**Test Case 6.5.2:** Considering the data provided (as given in Step 3), the output (thermal grid) is

**The thermal image for this case is:**



## Analysis:

### **Temperature Gradient on Left Side:**

The observed red shade on the left side aligns with the expectation of the maximum temperature on this side.

## Color Transition from Red to Green:

The observed transition from red to orange, to yellow, and finally to green as we move from the left border to the center corresponds to the anticipated thermal dissipation.

**Representation of Minimum Temperature on Right Side:**

The appearance of a blue shade on the right side aligns with the expectation of the minimum temperature on this side.

**Yellow Shade on Top Side:**

The presence of a yellow shade on the top side is consistent with the anticipation of a moderately high temperature.

**Dark Green Shade on Bottom Side:**

The observed dark green shade on the bottom side corresponds to the expectation of a slightly high temperature but less than the left and top sides.

**Net Light Shade of Green in the Middle-Left Region:**

The overall light shade of green in the middle-left region supports the prediction of color merging due to thermal dissipation.

**Extended Blue Region:**

The extension of the blue region from the right border to the center aligns with the absence of constant stimulation and also due to the lower temperatures of the bottom and right sides.

, which is in agreement with the test case expected output.

**Therefore, we can conclude that the program is functioning correctly.**

## Sample Outputs –

1.1)

```
Enter your choice (1-6): 1  
Enter the number of rows of the grid:
```

1.2)

```
Enter your choice (1-6): 2  
Enter the initial temperature of the top side:
```

1.3)

```
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation :
```

1.4)

```
Enter your choice (1-6): 4  
Enter the tolerance value:
```

1.5) Prints Thermal Data depending upon the Input.

1.6)

```
Enter your choice (1-6): 6  
End of the program!
```

1.7)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): @  
Invalid Choice! Please enter either 1, 2, 3, 4, 5, or 6:
```

1.8)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 165  
Invalid Choice! Please enter either 1, 2, 3, 4, 5, or 6:
```

2.1)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 15  
Enter the number of columns of the grid: 15
```

2.2)

```
Enter the number of rows of the grid: -2  
Please enter a positive integer value, greater than 3, for number of rows: 5  
Enter the number of columns of the grid: 6
```

2.3)

```
Enter your choice (1-6): 1
Enter the number of rows of the grid: 2
Please enter a positive integer value, greater than 3, for number of rows: 5
Enter the number of columns of the grid: 2
Please enter a positive integer value, greater than 3, for number of columns: 6
```

2.4)

```
Enter your choice (1-6): 1
Enter the number of rows of the grid: 50
Enter the number of columns of the grid: apple
Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive integer value, greater than 3, for number of columns: 4
```

2.5)

```
Enter the number of rows of the grid: -
Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive integer value, greater than 3, for number of rows: 4
Enter the number of columns of the grid: 4
```

3.1)

```
Enter the initial temperature of the top side: 10
Enter the initial temperature of the bottom side: 20
Enter the initial temperature of the left side: 30
Enter the initial temperature of the right side: 40
```

3.2)

```
Enter the initial temperature of the top side: @
Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side:
```

3.3)

```
Enter the initial temperature of the top side: @
Error! Invalid Input! Please enter a valid numeric input!

Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side: -5
Please enter a positive value, between 0 and 255 (inclusive), for temperature of the top side: 6
Enter the initial temperature of the bottom side:
```

3.4)

```
Enter the initial temperature of the top side: 169
Enter the initial temperature of the bottom side: 48
Enter the initial temperature of the left side: 256
Please enter a positive value, between 0 and 255 (inclusive), for temperature of the left side:
```

3.5)

```
Enter your choice (1-6): 5
10.00 10.00 10.00 10.00
30.00 0.00 0.00 0.00 40.00
30.00 0.00 0.00 0.00 40.00
30.00 0.00 0.00 0.00 40.00
30.00 0.00 0.00 0.00 40.00
20.00 20.00 20.00 20.00 20.00
```

4.1)

```
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 3  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 3  
Enter the stimulation value: 255  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1
```

4.2)

```
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : -5  
Please enter a value between 1 and (number of columns - 2), both inclusive, for the x-coordinate:
```

4.3)

```
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : -5  
Please enter a value between 1 and (number of columns - 2), both inclusive, for the x-coordinate: 3  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 7  
Please enter a value between 1 and (number of rows - 2), both inclusive, for the y-coordinate: 2  
Enter the stimulation value:
```

4.4)

```
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 3  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 3  
Enter the stimulation value: @  
Error! Invalid Input! Please enter a valid numeric input!  
  
Please enter a positive value, between 0 and 255 (inclusive), for the stimulation value:
```

4.5)

```
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 3  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 3  
Enter the stimulation value: 100  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 4  
Please enter either 0 or 1:
```

4.6)

```
Enter your choice (1-6): 5  
10.00 10.00 10.00 10.00 10.00  
30.00 0.00 0.00 0.00 40.00  
30.00 0.00 0.00 0.00 40.00  
30.00 0.00 0.00 100.00 40.00  
30.00 0.00 0.00 0.00 40.00  
20.00 20.00 20.00 20.00 20.00
```

5.1)

```
Enter your choice (1-6): 4  
Enter the tolerance value: 1
```

5.2)

```
Enter the tolerance value: !  
Error! Invalid Input! Please enter a valid numeric input!  
  
Please enter a positive value, between 0 and 255 (inclusive), for the tolerance value:
```

### 5.3)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 5  
10.00 10.00 10.00 10.00 10.00  
30.00 24.00 26.43 31.88 40.00  
30.00 32.37 41.02 53.20 40.00  
30.00 35.57 53.27 100.00 40.00  
30.00 29.82 37.60 48.97 40.00  
20.00 20.00 20.00 20.00 20.00
```

### 5.4)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 5  
10.00 10.00 10.00 10.00 10.00  
30.00 20.34 19.60 24.92 40.00  
30.00 24.41 24.37 29.91 40.00  
30.00 25.82 26.17 30.33 40.00  
30.00 24.15 24.18 27.88 40.00  
20.00 20.00 20.00 20.00 20.00
```

### 6.1.1)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 15  
Enter the number of columns of the grid: 15  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 2  
Enter the initial temperature of the top side: 255  
Enter the initial temperature of the bottom side: 100  
Enter the initial temperature of the left side: 10  
Enter the initial temperature of the right side: 50  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 8  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 8  
Enter the stimulation value: 255  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1
```

```

-----  

THERMAL DISTRIBUTION SIMULATION  

-----  

1. Create the Dynamic 2D Grid  

2. Set Boundary Conditions for the Grid  

3. Stimulate the Grid at a Specific Point  

4. Calculate the Thermal Distribution  

5. Print the Thermal Grid  

6. Exit  

Enter your choice (1-6): 4  

Enter the tolerance value: 1  

-----  

THERMAL DISTRIBUTION SIMULATION  

-----  

1. Create the Dynamic 2D Grid  

2. Set Boundary Conditions for the Grid  

3. Stimulate the Grid at a Specific Point  

4. Calculate the Thermal Distribution  

5. Print the Thermal Grid  

6. Exit  

Enter your choice (1-6): 5  

255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00  

10.00 129.50 175.93 197.40 208.86 215.61 219.68 222.58 222.27 220.54 215.04 206.61 188.27 150.19 50.00  

10.00 79.85 125.54 153.19 170.92 182.36 188.91 193.38 193.03 189.85 181.44 168.15 145.05 107.50 50.00  

10.00 56.51 94.35 121.23 141.72 155.16 164.07 170.01 170.37 166.74 156.58 141.79 118.79 87.49 50.00  

10.00 43.01 74.13 99.21 119.58 134.67 145.99 153.39 155.32 151.38 140.17 123.65 103.36 77.56 50.00  

10.00 35.35 61.12 84.42 105.22 121.74 135.75 145.64 148.54 143.30 130.29 113.11 93.44 71.91 50.00  

10.00 30.91 53.02 74.62 95.15 113.97 131.10 145.92 153.48 144.19 127.13 107.55 87.90 69.14 50.00  

10.00 28.89 48.71 68.33 88.10 109.16 131.12 155.90 179.00 155.38 128.75 105.73 85.02 66.75 50.00  

10.00 27.18 45.81 64.16 83.42 104.77 131.89 171.46 255.00 170.62 129.24 102.63 82.40 65.18 50.00  

10.00 26.31 44.37 61.52 79.18 98.16 120.21 145.52 168.29 144.03 118.46 97.15 79.40 63.61 50.00  

10.00 27.30 44.98 60.77 76.04 92.02 107.75 122.11 129.87 121.31 107.26 91.96 76.64 62.39 50.00  

10.00 30.31 48.65 62.93 75.91 88.48 99.00 108.12 111.33 107.82 98.48 88.05 76.23 62.66 50.00  

10.00 37.65 56.37 70.25 79.66 88.30 95.49 100.05 102.24 100.15 94.54 87.79 78.81 65.68 50.00  

10.00 55.11 72.79 82.05 87.86 92.11 95.88 98.25 99.67 98.20 95.33 92.17 85.55 74.76 50.00  

100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00

```

```

-----  

THERMAL DISTRIBUTION SIMULATION  

-----  

1. Create the Dynamic 2D Grid  

2. Set Boundary Conditions for the Grid  

3. Stimulate the Grid at a Specific Point  

4. Calculate the Thermal Distribution  

5. Print the Thermal Grid  

6. Exit  

Enter your choice (1-6): 6  

End of the program!

```

### 6.1.2)

```

-----  

THERMAL DISTRIBUTION SIMULATION  

-----  

1. Create the Dynamic 2D Grid  

2. Set Boundary Conditions for the Grid  

3. Stimulate the Grid at a Specific Point  

4. Calculate the Thermal Distribution  

5. Print the Thermal Grid  

6. Exit  

Enter your choice (1-6): 1  

Enter the number of rows of the grid: 15  

Enter the number of columns of the grid: 15  

-----  

THERMAL DISTRIBUTION SIMULATION  

-----  

1. Create the Dynamic 2D Grid  

2. Set Boundary Conditions for the Grid  

3. Stimulate the Grid at a Specific Point  

4. Calculate the Thermal Distribution  

5. Print the Thermal Grid  

6. Exit  

Enter your choice (1-6): 2  

Enter the initial temperature of the top side: 255  

Enter the initial temperature of the bottom side: 100  

Enter the initial temperature of the left side: 10  

Enter the initial temperature of the right side: 50

```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 8  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 8  
Enter the stimulation value: 255  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 1
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 5  
255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00 255.00  
10.00 128.25 174.09 194.68 204.54 209.95 212.88 214.67 214.80 213.43 208.67 201.18 184.75 148.40 50.00  
10.00 77.66 119.95 146.40 161.26 169.83 175.68 178.51 178.66 175.26 168.72 157.48 137.77 103.84 50.00  
10.00 52.44 85.44 109.72 125.57 136.02 142.73 146.29 146.07 142.78 136.07 124.78 107.23 82.64 50.00  
10.00 38.12 63.21 82.75 97.69 108.30 115.32 119.08 118.99 116.17 110.43 100.93 87.24 70.49 50.00  
10.00 29.25 47.69 63.95 77.66 87.71 94.74 98.17 98.47 95.88 91.34 83.77 73.63 62.10 50.00  
10.00 23.62 38.12 51.31 62.46 71.35 77.77 81.61 82.11 80.80 77.44 72.13 65.12 57.47 50.00  
10.00 20.84 32.40 43.29 53.17 61.13 65.82 69.65 71.52 70.34 67.31 63.25 58.47 53.69 50.00  
10.00 18.63 28.70 38.84 47.04 54.17 58.53 62.14 64.00 64.34 61.97 58.64 55.62 52.76 50.00  
10.00 18.62 28.51 37.51 45.80 51.25 55.80 59.01 60.61 61.06 60.22 57.45 55.27 51.74 50.00  
10.00 20.80 31.46 40.79 48.65 53.01 57.05 60.10 62.11 62.93 61.72 59.48 56.27 52.83 50.00  
10.00 25.42 38.26 48.09 55.01 58.76 62.12 64.66 66.03 66.85 65.47 63.80 60.03 55.59 50.00  
10.00 33.81 49.39 59.33 65.79 68.84 71.49 73.08 74.08 74.22 73.29 71.47 68.15 60.75 50.00  
10.00 52.69 69.68 76.64 81.01 82.87 83.98 85.06 85.30 85.04 84.36 83.06 80.37 72.78 50.00  
100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00 100.00
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 6  
End of the program!
```

### 6.2.1)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 25  
Enter the number of columns of the grid: 40  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 2  
Enter the initial temperature of the top side: 200  
Enter the initial temperature of the bottom side: 200  
Enter the initial temperature of the left side: 200  
Enter the initial temperature of the right side: 200  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 10  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 10  
Enter the stimulation value: 150  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 1.25
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 5  
200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00 200.00  
200.00 195.63 193.23 190.83 188.23 185.63 181.99 180.47 179.00 178.55 179.77 179.75 179.78 179.85 179.88 179.82 179.80 179.78 179.72 179.66 179.69 179.61 179.58 179.56 179.53 179.50 179.47 179.43 179.41 179.38 179.35 179.32 179.29 179.25 179.22 179.19 179.15 179.11 179.07 179.03 179.00 179.00  
200.00 197.20 177.04 169.23 162.98 155.98 158.86 147.11 146.88 141.74 139.29 138.87 139.76 132.49 139.41 138.32 136.64 124.47 123.26 122.23 122.78 121.79 122.35 123.58 124.62 136.05 136.73 129.28 138.98 138.38 137.76 134.16 134.69 146.08 151.21 137.95 136.13 176.35 137.16 200.00  
200.00 188.15 169.79 157.56 148.56 138.56 131.86 130.65 122.40 118.75 119.77 111.41 109.42 109.42 98.55 92.45 88.21 86.15 80.31 83.83 83.86 83.04 84.42 86.31 87.22 88.95 90.46 91.94 90.79 97.39 182.53 180.65 114.29 120.05 139.34 139.29 131.75 166.86 131.24 200.00  
200.00 188.09 169.08 156.65 138.79 138.94 122.27 116.86 111.19 108.83 103.89 98.51 96.91 91.26 76.13 78.30 65.82 62.97 59.65 56.92 56.76 58.43 68.43 62.27 66.89 68.49 73.86 79.25 87.66 94.07 106.03 116.94 127.75 132.92 158.99 178.85 200.00  
200.00 179.51 162.08 150.81 145.81 135.81 125.81 122.03 119.22 115.98 111.69 108.85 102.52 77.01 69.01 65.84 60.84 56.94 53.35 51.55 43.86 47.76 53.36 58.23 66.55 71.42 83.77 93.16 106.24 120.19 137.97 155.36 179.85 200.00  
200.00 178.81 158.81 142.43 132.43 122.43 115.22 115.98 111.69 108.85 102.77 100.55 94.87 90.23 74.47 63.55 58.84 54.95 50.35 46.88 41.55 47.76 53.36 58.23 66.55 71.42 83.77 93.16 106.24 120.19 137.97 155.36 179.85 200.00  
200.00 177.22 159.66 142.22 127.88 119.88 108.22 102.86 98.13 95.95 91.94 81.00 79.69 59.92 52.86 45.92 39.76 35.86 32.39 36.47 26.18 26.90 29.63 27.63 38.76 38.36 32.69 36.41 41.69 47.81 54.97 63.58 70.17 86.12 98.51 114.18 132.29 152.29 176.66 200.00  
200.00 177.52 158.72 142.37 126.79 114.93 105.19 97.15 90.86 84.86 78.13 71.37 60.94 57.97 51.04 46.94 40.94 36.78 33.98 32.12 31.82 33.80 31.57 32.66 33.26 36.12 38.81 41.92 46.62 52.77 59.72 66.41 75.93 87.04 106.58 113.88 132.83 170.78 200.00  
200.00 177.96 161.43 145.36 139.73 139.22 149.25 140.61 92.48 86.55 88.69 73.49 68.59 63.57 58.89 54.62 51.59 48.58 46.99 46.89 43.68 43.84 48.59 44.72 45.47 47.99 51.38 54.38 58.23 63.54 69.56 76.44 85.12 95.28 108.49 121.34 137.62 155.88 176.66 200.00  
200.00 178.07 162.93 150.81 145.81 135.81 125.81 122.03 119.22 115.98 111.69 108.85 102.52 77.01 69.01 65.84 60.84 56.94 53.35 51.55 43.86 41.55 47.76 53.36 58.23 66.55 71.42 83.77 93.16 106.24 120.19 137.97 155.36 179.85 200.00  
200.00 177.41 158.41 142.41 132.41 122.41 115.21 115.96 111.67 108.82 103.88 98.51 96.91 91.26 76.13 78.30 65.87 62.97 59.65 56.92 56.76 58.43 68.43 62.27 66.89 68.49 73.86 79.25 87.66 94.07 106.03 116.94 127.75 132.92 158.99 178.85 200.00  
200.00 177.87 158.52 147.89 139.52 130.13 120.13 119.04 104.31 109.07 99.48 97.75 89.12 86.41 82.27 81.49 80.91 81.99 82.99 82.88 84.45 87.21 89.75 91.85 95.12 108.13 109.62 112.97 121.29 129.26 139.09 131.01 146.86 182.81 200.00  
200.00 177.91 158.56 147.91 139.54 130.14 120.14 119.05 104.32 109.08 99.49 97.76 89.13 86.42 82.28 81.49 80.92 81.99 82.99 82.88 84.45 87.21 89.75 91.85 95.12 108.13 109.62 112.97 121.29 129.26 139.09 131.01 146.86 182.81 200.00  
200.00 178.00 158.58 147.93 139.56 130.15 120.15 119.06 104.33 109.09 99.50 97.77 89.14 86.43 82.30 81.51 80.93 81.99 82.99 82.88 84.46 87.22 89.76 91.86 95.13 108.14 109.63 112.98 121.30 129.27 139.10 131.02 146.87 182.82 200.00  
200.00 178.04 158.52 147.97 139.58 130.17 120.17 119.08 104.34 109.11 99.51 97.78 89.15 86.44 82.31 81.52 80.94 81.99 82.99 82.88 84.46 87.22 89.76 91.86 95.13 108.14 109.63 112.98 121.30 129.27 139.10 131.02 146.87 182.82 200.00  
200.00 178.08 158.56 147.91 139.52 130.12 120.12 119.06 104.35 109.09 99.52 97.79 89.13 86.45 82.30 81.51 80.93 81.99 82.99 82.88 84.46 87.22 89.76 91.86 95.13 108.14 109.63 112.98 121.30 129.27 139.10 131.02 146.87 182.82 200.00  
200.00 178.12 158.50 147.85 139.46 130.06 120.06 119.00 104.36 109.03 99.53 97.80 89.14 86.46 82.31 81.52 80.94 81.99 82.99 82.88 84.46 87.22 89.76 91.86 95.13 108.14 109.63 112.98 121.30 129.27 139.10 131.02 146.87 182.82 200.00  
200.00 178.16 158.44 147.79 139.40 130.00 120.00 119.04 104.37 109.07 99.54 97.81 89.15 86.47 82.32 81.53 80.95 81.99 82.99 82.88 84.47 87.23 89.77 91.87 95.14 108.15 109.63 112.97 121.31 129.28 139.11 131.03 146.88 182.83 200.00  
200.00 178.20 158.38 147.73 139.34 130.04 120.04 119.08 104.38 109.01 99.55 97.82 89.16 86.48 82.33 81.54 80.96 81.99 82.99 82.88 84.48 87.23 89.77 91.87 95.14 108.15 109.63 112.97 121.31 129.28 139.11 131.03 146.88 182.83 200.00  
200.00 178.24 158.32 147.67 139.28 130.08 120.08 119.12 104.39 109.05 99.56 97.83 89.17 86.49 82.34 81.55 80.97 81.99 82.99 82.88 84.49 87.24 89.78 91.88 95.15 108.16 109.64 112.98 121.32 129.29 139.12 131.04 146.89 182.84 200.00  
200.00 178.28 158.26 147.61 139.22 130.12 120.12 119.16 104.40 109.09 99.57 97.84 89.18 86.50 82.35 81.56 80.98 81.99 82.99 82.88 84.50 87.25 89.79 91.89 95.16 108.17 109.65 112.99 121.33 129.30 139.13 131.05 146.90 182.85 200.00  
200.00 178.32 158.20 147.55 139.16 130.16 120.16 119.20 104.41 109.13 99.58 97.85 89.19 86.51 82.36 81.57 80.99 81.99 82.99 82.88 84.51 87.26 89.80 91.90 95.17 108.18 109.66 112.10 121.34 129.31 139.14 131.06 146.91 182.86 200.00  
200.00 178.36 158.14 147.49 139.10 130.20 120.20 119.24 104.42 109.17 99.59 97.86 89.20 86.52 82.37 81.58 81.00 82.00 82.99 82.88 84.52 87.27 89.81 91.91 95.18 108.19 109.67 112.14 121.35 129.32 139.15 131.07 146.92 182.87 200.00  
200.00 178.40 158.08 147.43 139.04 130.24 120.24 119.28 104.43 109.21 99.60 97.87 89.21 86.53 82.38 81.59 81.01 82.01 82.99 82.88 84.53 87.28 89.82 91.92 95.19 108.20 109.68 112.18 121.36 129.33 139.16 131.08 146.93 182.88 200.00  
200.00 178.44 158.02 147.37 139.08 130.28 120.28 119.32 104.44 109.25 99.61 97.88 89.22 86.54 82.39 81.60 81.02 82.02 82.99 82.88 84.54 87.29 89.83 91.93 95.20 108.21 109.69 112.22 121.37 129.34 139.17 131.09 146.94 182.89 200.00  
200.00 178.48 157.96 147.31 139.02 130.32 120.32 119.36 104.45 109.29 99.62 97.89 89.23 86.55 82.40 81.61 81.03 82.03 82.99 82.88 84.55 87.30 89.84 91.94 95.21 108.22 109.70 112.26 121.41 129.35 139.18 131.10 146.95 182.90 200.00  
200.00 178.52 157.90 147.25 139.06 130.36 120.36 119.40 104.46 109.33 99.63 97.90 89.24 86.56 82.41 81.62 81.04 82.04 82.99 82.88 84.56 87.31 89.85 91.95 95.22 108.23 109.71 112.30 121.45 129.36 139.19 131.11 146.96 182.91 200.00  
200.00 178.56 157.84 147.19 139.10 130.40 120.40 119.44 104.47 109.37 99.64 97.91 89.25 86.57 82.42 81.63 81.05 82.05 82.99 82.88 84.57 87.32 89.86 91.96 95.23 108.24 109.75 112.34 121.49 129.37 139.20 131.12 146.97 182.92 200.00  
200.00 178.60 157.78 147.13 139.14 130.44 120.44 119.48 104.48 109.41 99.65 97.92 89.26 86.58 82.43 81.64 81.06 82.06 82.99 82.88 84.58 87.33 89.87 91.97 95.24 108.25 109.79 112.38 121.53 129.38 139.21 131.13 146.98 182.93 200.00  
200.00 178.64 157.72 147.07 139.18 130.48 120.48 119.52 104.49 109.45 99.66 97.93 89.27 86.59 82.44 81.65 81.07 82.07 82.99 82.88 84.59 87.34 89.88 91.98 95.25 108.26 109.83 112.42 121.57 129.39 139.22 131.14 146.99 182.94 200.00  
200.00 178.68 157.66 147.01 139.22 130.52 120.52 119.56 104.50 109.49 99.67 97.94 89.28 86.60 82.45 81.66 81.08 82.08 82.99 82.88 84.60 87.35 89.89 91.99 95.26 108.27 109.87 112.46 121.61 129.40 139.23 131.15 146.10 182.95 200.00  
200.00 178.72 157.60 146.95 139.26 130.56 120.56 119.60 104.51 109.53 99.68 97.95 89.30 86.61 82.46 81.67 81.09 82.09 82.99 82.88 84.61 87.36 89.90 92.00 95.27 108.28 109.91 112.50 121.65 129.41 139.24 131.16 146.11 182.96 200.00  
200.00 178.76 157.54 146.89 139.30 130.60 120.60 119.64 104.52 109.57 99.69 97.96 89.31 86.62 82.47 81.68 81.10 82.10 82.99 82.88 84.62 87.37 89.91 92.01 95.28 108.29 109.95 112.54 121.70 129.42 139.25 131.17 146.12 182.97 200.00  
200.00 178.80 157.48 146.83 139.34 130.64 120.64 119.68 104.53 109.61 99.70 97.97 89.32 86.63 82.48 81.69 81.11 82.11 82.99 82.88 84.63 87.38 89.92 92.02 95.29 108.30 109.99 112.58 121.74 129.43 139.26 131.18 146.13 182.98 200.00  
200.00 178.84 157.42 146.77 139.38 130.68 120.68 119.72 104.54 109.65 99.71 97.98 89.33 86.64 82.49 81.70 81.12 82.12 82.99 82.88 84.64 87.39 89.93 92.03 95.30 108.31 109.10 112.62 121.78 129.44 139.27 131.19 146.14 182.99 200.00  
200.00 178.88 157.36 146.71 139.42 130.72 120.72 119.76 104.55 109.69 99.72 97.99 89.34 86.65 82.50 81.71 81.13 82.13 82.99 82.88 84.65 87.40 89.94 92.04 95.31 108.32 109.14 112.66 121.82 129.45 139.28 131.20 146.15 182.100.00  
200.00 178.92 157.30 146.65 139.46 130.76 120.76 119.80 104.56 109.73 99.73 98.00 89.35 86.66 82.51 81.72 81.14 82.14 82.99 82.88 84.66 87.41 89.95 92.05 95.32 108.33 109.18 112.70 121.86 129.46 139.29 131.21 146.16 182.100.00  
200.00 178.96 157.24 146.59 139.50 130.80 120.80 119.84 104.57 109.77 99.74 98.01 89.36 86.67 82.52 81.73 81.15 82.15 82.99 82.88 84.67 87.42 89.96 92.06 95.33 108.34 109.22 112.74 121.90 129.47 139.30 131.22 146.17 182.100.00  
200.00 179.00 157.18 146.53 139.54 130.84 120.84 119.88 104.58 109.81 99.75 98.02 89.37 86.68 82.53 81.74 81.16 82.16 82.99 82.88 84.68 87.43 89.97 92.07 95.34 108.35 109.26 112.78 121.94 129.48 139.31 131.23 146.18 182.100.00  
200.00 179.04 157.12 146.47 139.58 130.88 120.88 119.92 104.59 109.85 99.76 98.03 89.38 86.69 82.54 81.75 81.17 82.17 82.99 82.88 84.69 87.44 89.98 92.08 95.35 108.36 109.28 112.82 121.98 129.49 139.32 131.24 146.19 182.100.00  
200.00 179.08 157.06 146.41 139.62 130.92 120.92 119.96 104.60 109.89 99.77 98.04 89.39 86.70 82.55 81.76 81.18 82.18 82.99 82.88 84.70 87.45 89.99 92.09 95.36 108.37 109.32 112.86 122.02 129.50 139.33 131.25 146.20 182.100.00  
200.00 179.12 157.00 146.35 139.66 130.96 120.96 119.10 104.61 109.93 99.78 98.05 89.40 86.71 82.56 81.77 81.19 82.19 82.99 82.88 84.71 87.46 89.10 92.10 95.37 108.38 109.36 112.90 122.06 129.51 139.34 131.26 146.21 182.100.00  
200.00 179.16 156.94 146.29 139.70 130.10 120.10 119.14 104.62 109.97 99.79 98.06 89.41 86.72 82.57 81.78 81.20 82.20 82.99 82.88 84.72 87.47 89.11 92.11 95.38 108.39 109.40 112.94 122.10 129.52 139.35 131.27 146.22 182.100.00  
200.00 179.20 156.88 146.23 139.74 130.14 120.14 119.18 104.63 109.10 99.80 98.07 89.42 86.73 82.58 81.79 81
```

### 6.2.2)

---

## THERMAL DISTRIBUTION SIMULATION

---

```
THERMAL DISTRIBUTION SIMULATION
-----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 4
Enter the tolerance value: 1.25
```

### 6.3.1)

```
THERMAL DISTRIBUTION SIMULATION
-----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 1
Enter the number of rows of the grid: 15
Enter the number of columns of the grid: 15

-----
THERMAL DISTRIBUTION SIMULATION
-----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 2
Enter the initial temperature of the top side: 100
Enter the initial temperature of the bottom side: 20
Enter the initial temperature of the left side: 20
Enter the initial temperature of the right side: 20

-----
THERMAL DISTRIBUTION SIMULATION
-----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 3
Please note that 0 represents the first column and
Enter the x-coordinate (columns - left to right) p
Enter the y-coordinate (rows - top to bottom) p
Enter the stimulation value: 100
Do you wish to keep the stimulation temperature co
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 6  
End of the program!
```

### 6.3.2)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 15  
Enter the number of columns of the grid: 15  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 2  
Enter the initial temperature of the top side: 100  
Enter the initial temperature of the bottom side: 100  
Enter the initial temperature of the left side: 200  
Enter the initial temperature of the right side: 200  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 7  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 7  
Enter the stimulation value: 100  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 1
```

---

**Thermal Distribution Simulation**

---

1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit

Enter your choice (1-6): 5

100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	
200.00	146.81	124.51	113.32	107.99	104.81	103.30	102.10	102.01	104.80	108.03	113.80	124.10	146.89	200.00					
200.00	163.98	148.19	124.36	114.98	109.19	106.29	104.32	105.05	109.18	114.81	124.33	139.44	163.47	200.00					
200.00	171.38	149.16	132.68	121.90	114.22	109.58	107.50	108.35	113.35	120.26	131.78	148.32	171.25	200.00					
200.00	178.74	154.71	132.93	125.73	117.39	112.68	110.32	111.36	116.98	124.96	136.51	153.32	174.41	200.00					
200.00	177.62	157.86	141.67	129.03	119.68	114.70	112.49	113.66	119.18	127.39	139.73	156.27	176.56	200.00					
200.00	178.66	159.67	142.99	130.05	120.94	115.49	113.77	115.22	121.23	129.56	141.52	157.89	177.87	200.00					
200.00	178.59	159.14	142.82	131.85	121.66	116.48	114.99	115.96	122.17	130.62	142.80	158.33	178.33	200.00					
200.00	177.73	157.93	141.85	129.66	120.73	115.42	113.75	115.33	120.89	130.40	141.96	157.91	178.27	200.00					
200.00	176.71	156.92	140.88	128.63	128.61	114.56	112.87	114.31	119.39	128.13	140.58	156.90	176.86	200.00					
200.00	174.59	153.43	137.24	125.43	117.44	112.67	111.15	112.82	116.72	124.69	136.56	153.57	174.68	200.00					
200.00	171.82	148.77	132.86	120.83	112.92	108.65	107.44	109.09	113.44	119.92	131.27	148.68	170.67	200.00					
200.00	163.92	140.53	124.62	114.46	108.56	105.44	104.79	106.35	109.19	114.10	123.76	139.21	163.29	200.00					
200.00	146.93	124.82	114.18	107.27	103.98	102.65	102.44	102.33	104.15	107.28	113.12	123.77	146.74	200.00					
100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	

---

**Thermal Distribution Simulation**

---

1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit

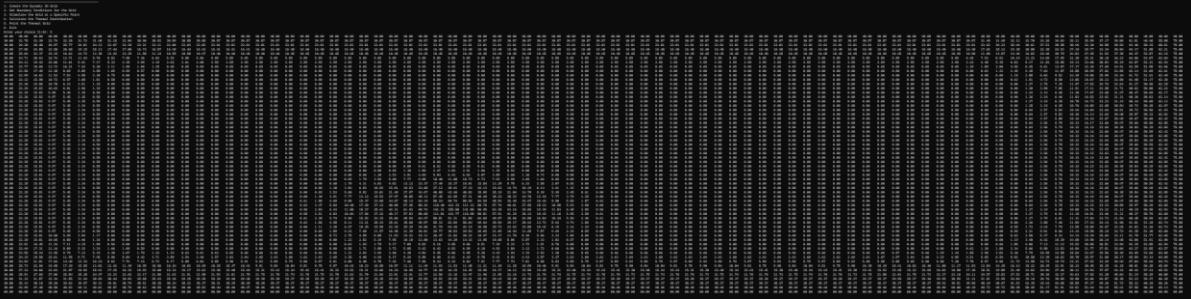
Enter your choice (1-6): 6

End of the program!

### 6.4.1)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 60  
Enter the number of columns of the grid: 80  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 2  
Enter the initial temperature of the top side: 40  
Enter the initial temperature of the bottom side: 60  
Enter the initial temperature of the left side: 30  
Enter the initial temperature of the right side: 75  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 30  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 40  
Enter the stimulation value: 255  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 0.5
```



```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 6  
End of the program!
```

#### 6.4.2)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 60  
Enter the number of columns of the grid: 80  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 2  
Enter the initial temperature of the top side: 40  
Enter the initial temperature of the bottom side: 60  
Enter the initial temperature of the left side: 30  
Enter the initial temperature of the right side: 75  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 30  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 40  
Enter the stimulation value: 255  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 0.5
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 6  
End of the program!
```

### 6.5.1)

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 1  
Enter the number of rows of the grid: 20  
Enter the number of columns of the grid: 15  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 2  
Enter the initial temperature of the top side: 150  
Enter the initial temperature of the bottom side: 75  
Enter the initial temperature of the left side: 200  
Enter the initial temperature of the right side: 40  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 3  
Please note that 0 represents the first column and 0 represents the first row.  
Enter the x-coordinate (columns - left to right) point of stimulation : 10  
Enter the y-coordinate (rows - top to bottom) point of stimulation : 7  
Enter the stimulation value: 185  
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 1
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 0.8
```

```

----- THERMAL DISTRIBUTION SIMULATION -----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 5
150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00
200.00 172.20 159.38 152.44 148.57 145.99 144.16 142.14 139.98 137.20 132.86 126.91 116.41 94.67 40.00
200.00 179.43 164.86 154.71 147.90 142.88 139.36 136.19 132.55 127.01 120.22 110.35 96.12 73.26 40.00
200.00 181.67 166.90 155.64 147.41 141.05 136.13 131.69 127.04 121.04 113.48 101.25 85.31 64.44 40.00
200.00 182.33 167.42 155.57 145.96 138.77 133.34 129.30 124.81 119.54 111.40 98.01 81.39 61.06 40.00
200.00 183.19 167.81 155.23 145.10 137.60 132.10 128.36 125.34 123.09 117.35 100.92 81.19 60.39 40.00
200.00 182.63 167.32 154.42 143.85 136.39 130.13 126.72 127.08 131.20 135.19 109.06 83.08 60.32 40.00
200.00 181.93 166.36 153.43 142.54 133.98 127.27 124.34 127.06 140.66 185.00 119.74 84.90 60.58 40.00
200.00 181.64 165.55 151.42 139.73 130.70 123.47 119.17 119.07 122.20 127.03 162.70 78.26 58.11 40.00
200.00 180.97 163.79 148.91 136.49 126.45 117.96 111.77 107.86 105.11 100.90 86.57 70.24 54.63 40.00
200.00 180.39 162.61 146.99 133.70 122.82 113.08 105.01 98.60 92.53 85.86 75.39 63.48 51.18 40.00
200.00 178.98 160.26 143.76 129.45 118.05 107.46 98.68 90.94 83.57 76.56 67.54 58.05 48.54 40.00
200.00 178.25 157.70 140.31 125.31 113.41 102.78 93.56 85.06 77.13 70.26 62.07 54.51 46.88 40.00
200.00 176.34 154.95 136.51 120.87 108.32 97.65 88.53 80.62 72.73 65.29 58.80 51.97 45.42 40.00
200.00 175.00 151.93 132.94 116.39 103.32 92.74 84.30 76.16 68.91 62.21 55.87 50.09 44.79 40.00
200.00 171.73 147.60 127.91 111.25 98.69 88.74 80.76 73.84 67.47 60.78 55.25 49.69 44.58 40.00
200.00 167.02 139.79 119.85 104.91 93.65 85.09 78.20 71.94 66.33 61.35 56.49 51.80 45.89 40.00
200.00 156.57 127.51 108.74 96.09 87.16 80.84 76.08 71.44 67.34 63.71 59.79 55.13 49.19 40.00
200.00 134.62 106.91 93.49 86.45 81.22 77.26 74.91 72.26 70.73 68.31 65.72 62.64 56.71 40.00
75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00

----- THERMAL DISTRIBUTION SIMULATION -----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 6
End of the program!

```

## 6.5.2)

```

----- THERMAL DISTRIBUTION SIMULATION -----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 1
Enter the number of rows of the grid: 20
Enter the number of columns of the grid: 15

----- THERMAL DISTRIBUTION SIMULATION -----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 2
Enter the initial temperature of the top side: 150
Enter the initial temperature of the bottom side: 75
Enter the initial temperature of the left side: 200
Enter the initial temperature of the right side: 40

----- THERMAL DISTRIBUTION SIMULATION -----
1. Create the Dynamic 2D Grid
2. Set Boundary Conditions for the Grid
3. Stimulate the Grid at a Specific Point
4. Calculate the Thermal Distribution
5. Print the Thermal Grid
6. Exit
Enter your choice (1-6): 3
Please note that 0 represents the first column and 0 represents the first row.
Enter the x-coordinate (columns - left to right) point of stimulation : 10
Enter the y-coordinate (rows - top to bottom) point of stimulation : 7
Enter the stimulation value: 185
Do you wish to keep the stimulation temperature constant?, Please enter 1 for YES or 0 for NO: 0

```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 4  
Enter the tolerance value: 0.8  
  
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 5  
150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00 150.00  
200.00 171.18 158.68 151.17 146.14 142.49 139.77 137.24 134.19 131.89 128.15 122.65 112.85 92.59 40.00  
200.00 178.09 162.37 151.04 142.68 135.88 130.33 125.02 120.34 116.17 109.89 101.52 89.23 69.61 40.00  
200.00 179.82 163.83 150.79 139.48 130.82 122.52 115.84 108.81 102.56 95.67 86.23 73.90 58.48 40.00  
200.00 180.27 163.28 148.81 136.51 125.40 115.74 106.77 99.36 91.53 84.00 74.84 64.43 52.37 40.00  
200.00 180.06 162.19 146.61 133.32 121.46 110.18 100.61 91.54 83.32 74.96 66.63 57.57 48.33 40.00  
200.00 179.81 160.87 144.18 129.70 116.93 105.51 94.55 85.50 77.24 68.79 61.05 53.42 45.67 40.00  
200.00 179.34 159.41 142.19 127.43 113.96 101.39 90.35 80.60 72.34 63.87 56.16 49.42 43.95 40.00  
200.00 178.12 158.21 140.79 124.82 111.11 98.61 87.55 77.11 68.58 60.17 53.21 47.26 42.76 40.00  
200.00 177.76 157.40 138.94 122.99 108.99 96.38 85.13 74.62 65.67 57.91 51.18 45.58 41.92 40.00  
200.00 177.43 156.84 137.53 121.23 106.47 93.77 82.95 72.55 63.53 56.56 49.94 44.91 41.38 40.00  
200.00 177.06 155.87 136.04 118.95 103.90 91.28 80.36 70.11 62.28 54.85 49.08 44.69 41.58 40.00  
200.00 175.85 153.55 133.71 116.64 101.93 89.07 78.12 68.19 60.64 54.45 48.82 44.17 41.22 40.00  
200.00 174.58 151.60 131.57 113.94 99.11 86.95 75.85 66.88 59.60 53.50 48.56 43.94 41.14 40.00  
200.00 173.66 149.28 128.03 110.40 96.55 84.78 74.46 65.83 59.38 53.37 48.98 44.90 41.45 40.00  
200.00 170.80 144.70 123.45 106.17 92.89 82.15 73.37 65.60 59.68 54.59 50.09 46.22 42.58 40.00  
200.00 165.71 137.82 117.01 100.85 88.65 79.54 72.32 66.49 61.99 57.17 53.54 49.22 44.58 40.00  
200.00 156.21 126.13 106.97 93.65 84.22 77.94 72.82 68.01 64.62 61.43 57.68 53.52 48.37 40.00  
200.00 133.98 106.58 93.30 84.64 79.53 76.13 73.02 71.01 68.95 67.13 65.16 61.80 56.29 40.00  
75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00 75.00
```

```
-----  
THERMAL DISTRIBUTION SIMULATION  
-----  
1. Create the Dynamic 2D Grid  
2. Set Boundary Conditions for the Grid  
3. Stimulate the Grid at a Specific Point  
4. Calculate the Thermal Distribution  
5. Print the Thermal Grid  
6. Exit  
Enter your choice (1-6): 6  
End of the program!
```

# User Guide

This program simulates thermal dissipation in a thin metal plate with constant (isothermal) temperatures on each side, given a simulation temperature and a point on the plate. The simulation calculates the thermal distribution across the system and converts it to a visual representation in a bitmap image file. The program ensures proper memory cleanup after execution.

## 1. Create the Dynamic 2D Grid (Option 1):

- When selected, the program prompts you to enter the number of rows and columns for the grid.
- If the grid has not been created before, a new grid is created. Otherwise, the previously created grid is deallocated before a new grid is created.
- Ensure you enter positive integers greater than 3.
- This step initializes the dynamic 2D grid.

## 2. Set Boundary Conditions for the Grid (Option 2):

- After creating the grid, set boundary temperatures for the top, bottom, left, and right sides.
- Enter temperatures between 0 and 255 (inclusive).
- The program sets these boundary conditions in the grid.

## 3. Stimulate the Grid at a Specific Point (Option 3):

- Once the grid and boundary conditions are set, stimulate the grid at a specific point.
- Enter the x and y coordinates (columns and rows, respectively) for the stimulation point.
- Input the stimulation value and choose whether the stimulation temperature should be constant or not.

## 4. Calculate the Thermal Distribution (Option 4):

- After creating the grid, setting boundary conditions, and stimulating the grid, calculate the thermal distribution.
- Input a tolerance value for equilibrium.
- The simulation loop iteratively calculates thermal dissipation until equilibrium is reached.
- The thermal distribution is then converted to a bitmap image and saved as "thermal\_distribution.bmp."

## 5. Display the Thermal Grid (Option 5):

- View the thermal grid after creating it.
- This option helps you visualize the initial state of the thermal distribution.

## 6. Exit the Program (Option 6):

- Choose this option to terminate the program.
- After exiting from the switch structure, memory cleanup is done before exiting the program in main.