Assignment 2: Mechanical Engineering Case Study – Thermal Distribution Simulation

Available: Monday November 6, 2023.
Due: Sunday November 19th @ 11:55pm (Submit via the Brightspace submission system)

Programming aspects to get familiarized with:

- Pointers,
- Dynamic memory allocation,
- Multidimensional arrays,
- Subprograms / Functions.

Problem Statement:

The temperature distribution in a thin metal plate with constant (isothermal) temperatures on each side can be modeled using a 2D grid, as shown in Figure 1. The number of points in the grid are not specified, as are the initial temperatures on the four sides. The temperatures of the interior points are initialized to zero, but they change according to the temperatures around them. Assuming that 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 labeled x in the grid. 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.

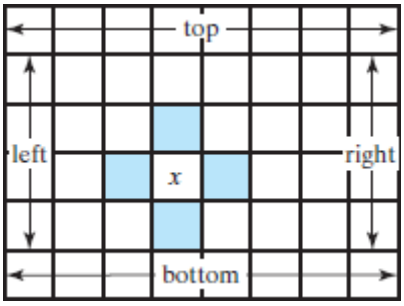

Figure 1: Thermal image distribution simulation.

Develop a software to simulate this temperature distribution for a grid with unknown dimensions (number of rows and columns). The program prompts the user for the dimensions of the thermal image (number of rows and number of columns). The program creates a dynamic 2D array to store the thermal data. The program prompts the user to enter the temperatures for the four sides (top, right, bottom, and left). Use a one-dimensional array to store the four sides initial temperatures.

The program must simulate the thermal distribution associated with the thermal stimulation of the grid. Let's assume that the thermal stimulation results in one point modified (one point of thermal stimulation). The new values are used to update the neighboring points. The program continues updating the points, moving across the rows until the temperature differences (between the previous

temperature and the newly calculated temperature) for all the points are less than a tolerance value (provided by the user). Once the simulation is completed, the resulting thermal distribution data must be stored in an image file. You may design this program using multiple functions, such as:

```
void create2DGrid(…);
void displayGrid (…);
void setBoundaryConditions(…);
void stimulateGrid(…);
void simulateThermalDissipation(…);
```

Develop a software solution, which will do the following:

- Present the user with menu selections for 1) Create the dynamic 2D grid 2) Set initial side temperature values for the grid, 3) Stimulate the grid at a specific point, 4) calculate the thermal distribution based on the specific stimulation and initial conditions (and save the results in an image file), 5) Print the thermal grid on the screen, and 6) Exit the program.
- Assume the input image shown in Figure 1 has unknown dimensions.
- Write a subprogram to generate the 2D dynamic grid. Your program should validate the user input and accommodate any dimensions for the image.
- Write another subprogram to initialize the side temperatures of the grid.
- Write a subprogram to stimulate the grid at a specific point.
- Write a subprogram to calculate the thermal distribution as a result of thermal stimulation and to save the resulting thermal distribution in an image file.
- Write a subprogram to display the 2D grid on the output screen.
- If the user selects the exit option, the program prints a terminating message and quits the program.
- Make sure to release any dynamically created memory when you no longer need it.
- You must use the auxiliary header file (`Bitmap_Helper.h`) to store the thermal data in the form of an image.
- You are not allowed to use advanced data structures for developing your solution.
- Make sure your code is well commented.
- Add any additional subprograms if you need to.
- You can make additional assumptions in case you feel any information is missing. Make sure to clearly state them.

Submission

Your submission **must** include an assignment report (Word Document or PDF) and a C++ source code (.cpp file). The report must follow the five steps model for software development (as discussed in class):

a) Step 1: Problem Identification and Statement (5 points)
b) Step 2: Gathering Information (10 points)
c) Step 3: Test Cases and algorithm (35 points)
d) Step 4: Code or implementation (35 points)
e) Step 5: Test and Verification (a minimum of 4 test cases) (15 points)