PROGRAM 11

QUESTION:

Write a program to enter a m x n matrix and print the sum of boundary elements.

Class Name – SumOfBoundaryElements

Instance variable:

* arr[][] – Integer array to store the input matrix.
* int s – to store the sum of boundary elements.

Methods:

* SumOfBoundaryElements (int mm) – to initialize input in the matrix.
* void input() – to take input in the Matrix.
* void outerSum() - to calculate sum of outer elements.
* void display() – to display the original matrix along with sum of outer elements.

ALGORITHM:

1. Initialize the class `SumOfBoundaryElement` with an integer `M` as a parameter, representing the number of rows and columns for the square matrix.
2. Inside the constructor:
   * + 1. Set `M` to the provided value.
       2. Initialize the sum `s` to 0.
       3. Create a 2D integer array `arr` with dimensions M x M.
3. Implement the `input` method:
   * + 1. Create a `Scanner` object `in` to read input from the console.
       2. Display a prompt to the user: "Enter the matrix elements."
       3. Use nested loops to iterate through each row and column of the matrix:
       4. Read an integer value from the user using `in.nextInt()`.
       5. Store the read value in the corresponding position of the `arr` 2D array.
4. Implement the `outerSum` method:
   * + 1. Initialize a variable `s` to 0 to represent the sum of boundary elements.
       2. Use nested loops to iterate through each row and column of the matrix:
       3. Check if the current element is on the boundary by verifying that it is on the first row, first column, last row, or last column.
       4. If the condition is met, add the value of the current element to `s`.

5. Implement the `display` method:

* + - 1. Display the message: "The original Matrix is : "
      2. Use nested loops to iterate through each row and column of the matrix:
      3. Print the value of the current element followed by a space.
      4. Print a newline character after each row.
      5. Display the message: "The boundary elements sum is " followed by the value of `s`.

1. In the `main` method:
   * + 1. Create a `Scanner` object `sc` to read input from the console.
       2. Display a prompt to the user: "Enter the number of rows."
       3. Read an integer `n` from the user to represent the number of rows (and columns) of the square matrix.
       4. Create an instance of the `SumOfBoundaryElement` class with `n` as the argument.
       5. Call the `input` method to input matrix elements.
       6. Call the `outerSum` method to compute the sum of boundary elements.
       7. Call the `display` method to display the original matrix and the sum of boundary elements.

**VARIABLE DESCRIPTION TABLE**

|  |  |  |
| --- | --- | --- |
| Variable | Data Type | Variable Description |
| Name |  |  |
| arr | int[][] | 2D integer array |
|  |  | representing the matrix. |
| s | int | Integer variable to store |
|  |  | the sum of elements. |
| M | int | Integer representing the |
|  |  | number of rows/columns |
|  |  | in the matrix. |
| mm | int | Integer parameter passed |
|  |  | to the constructor to set |
|  |  | the matrix size. |
| in | Scanner | Scanner object for |
|  |  | reading input from the |
|  |  | console. |
| sc | Scanner | Scanner object for |
|  |  | reading input from the |
|  |  | console in the `main` |
|  |  | method. |
| n | int | Integer representing the |
|  |  | number of rows/columns |
|  |  | in the matrix (user input). |
| obj | SumOfBoundaryElement | Instance of the |
|  |  | `SumOfBoundaryElement` |
|  |  | class. |