|  |  |  |
| --- | --- | --- |
|  | **2D REGULAR SHAPES FROM IRREGULAR POLYGON SHAPES** | Rev. No. 2  25-Nov-23 |

1. **Problem Statement/Objective**

Identify the usable 2D regular shapes from the given irregular polygonal structural layout by excluding the non-utilizable spaces marked as irregular polygon space.

1. **Major Stages of Development**

Stage 1: Receive the layout and the exclusion list of shapes from the user

Stage 2: Identify the usable 2D geometrical shapes of usable area

Stage 3: Find the maximum no. of usable rectangle shapes of size 2X2 from the usable area

1. **Stage 1**
2. **User Story**

As a user of the system, I shall be allowed to upload the list of end points denoting the irregular structural layout. In addition, I shall also be allowed to upload list of end points corresponding to the exclusion list of irregular polygonal space. The number of exclusions accepted shall be max. 2 spaces.

1. **Acceptance Criteria**

* Validate the end points of irregular structural layout by ensuring that they form a closed object.
* Similarly, validate the exclusion end points to form a closed object. Also, the closed object should lie inside the irregular structural layout.

**Algorithm in Detail:**

// Polygon with (x,y) coordinates

int[][] layout = new int[][]{ {37,78}, {149,78}, {149,12}, {35, 12}, {35, 78}, {37, 78}};

// Exclusion polygons with (x,y) coordinates

input - No of Polygons - 2

Exclusion 1 = new int[][]{ {49,68}, {74,23}, {74,68}, {49, 68}};

Exclusion 2 = new int[][]{ {88,68}, {88,46}, {138,46}, {138, 68}, {88, 68}};

Step 0.: Ensure the above inputs are received from a flat file which shall be read and parsed by the program

Step 1. Check if the layout and the exclusions are closed polygon

Step 2. Find the bottom left and top right coordinates of the layout to get the boundary of the layout

Step 3. Check the exclusions are mentioned with in the layout.

Step 4. Decide on the scale factor of the boundary to construct the 2D array

149-35 = 114, 78-12=66 - (144, 66) Layout

74-49 = 25, 68-23 = 45 - (25,45) Exclusion 1

138-88 = 50, 68-46 = 22 - (50,22) Exclusion 1

HCF(144,25,50) = ? (Scaling Factor of X)

HCF (66,45,22) = ? (Scaling Factor of Y)

Step 5. Construct a 2D array using the boundary and scaling factor obtained in Step 2

Step 6: Assign default '1' as default to all positions

Step 7: Assign '0' for coordinates given in Exclusion array

This concludes the validation and layout formation for the given input

1. **Stage 2**
2. **User Story**

The usable 2D regular shape shall be shown for analysis and confirmation

1. **Acceptance Criteria**

Remove the exclusion shapes and display the usable shape of the given irregular polygonal structure

1. **Stage 3**
2. **User Story**

Traverse the scaled layout array after exclusion, to find ‘n’ number of sub layouts of size 2X2 where ‘n’ is the maximum possible sub layouts possible

**Acceptance Criteria**

1. Traverse from top left to find the first ‘1’
   1. Check for any possible 2 X2 array from the location
   2. Mark as assigned if available.
2. Repeat the above process from left to right
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
3. Repeat Step 1 & 2, by traversing from right to left
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases
4. Repeat Step 1 & 2, by traversing from bottom left to right
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases
5. Repeat Step 1 & 2, by traversing from bottom right to left
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases
6. Repeat Step 1 & 2, by traversing from top left to down
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases
7. Repeat Step 1 & 2, by traversing from top right to down
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases
8. Repeat Step 1 & 2, by traversing from bottom left to top
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases
9. Repeat Step 1 & 2, by traversing from bottom right to top
   1. Count the no. of sub layouts received
   2. Get the uncovered layouts
   3. Ignore / Cancel the operation if any of the following condition fails
      1. No. Of Layouts are less than any of the quantities obtained earlier
      2. Uncovered layouts are greater than any of the existing cases