

Generalized Voronoi Path Finding Algorithm

AKSHAY BHOLA 14060

Content

- ▶ Introduction
- ▶ Mathematical Defination
- ▶ Roadmap
- ▶ Implementation
- ▶ Test Cases

Introduction

- ▶ Partitioning of plane into regions based on distance of points in space to obstacles in space
- ▶ Regions in cell correspond to a particular obstacle, with minimum distance with that obstacle
- ▶ The **Roadmap** generated by the boundaries of the cell correspond to the path with least proximity to obstacle
- ▶ The starting and end point are linked to the roadmap through which major path traversal is carried out

Mathematical Definition

- ▶ The formal mathematical definition of voronoi cells are

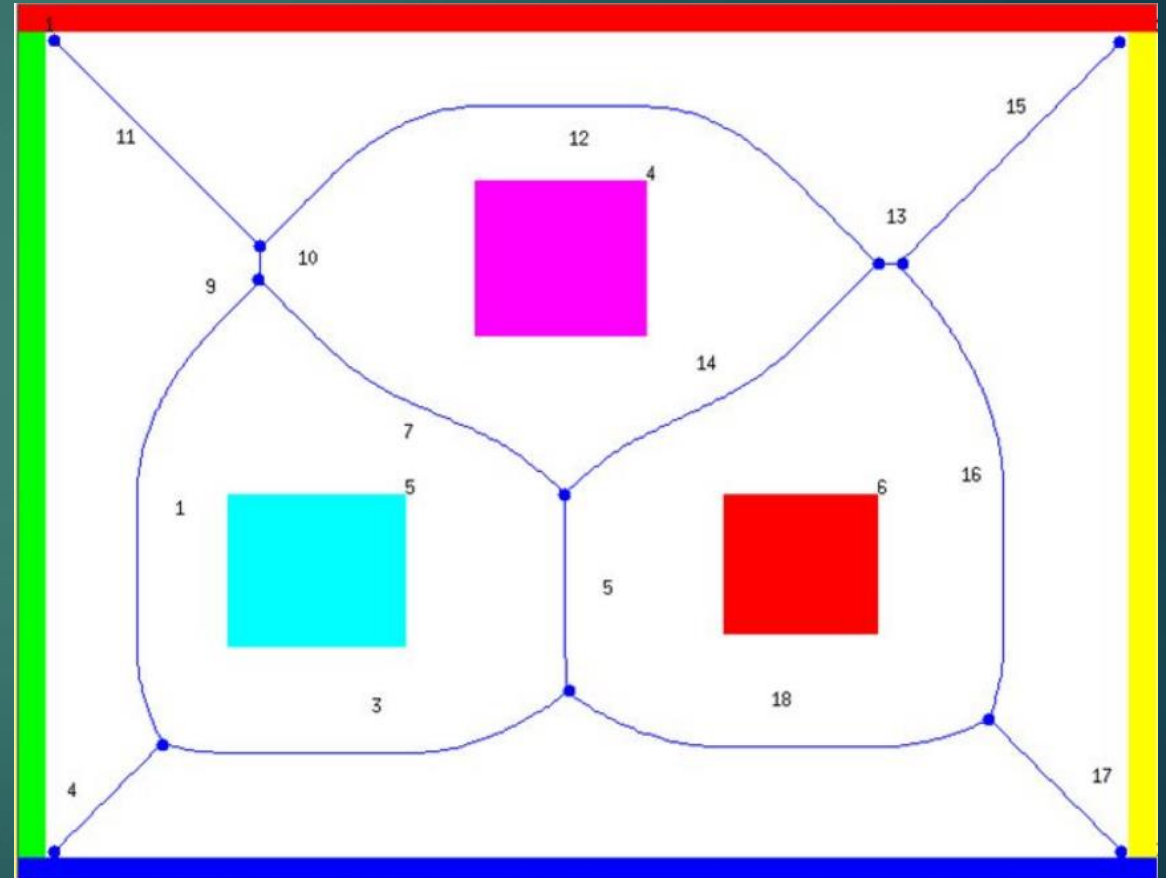
$$R_k = \{x \in X \mid d(x, P_k) < d(x, P_j) \text{ for all } j \neq k\}$$

- ▶ Similarly, the definition of the generated roadmap are:

$$A = \{x \in X \mid d(x, P_k) = d(x, P_j) \text{ for all } j \neq k\}$$

Roadmap

- ▶ If path exist, than this method assurly provides path
- ▶ Unlike other method, this method doesn't involve scaling factor adjustment.

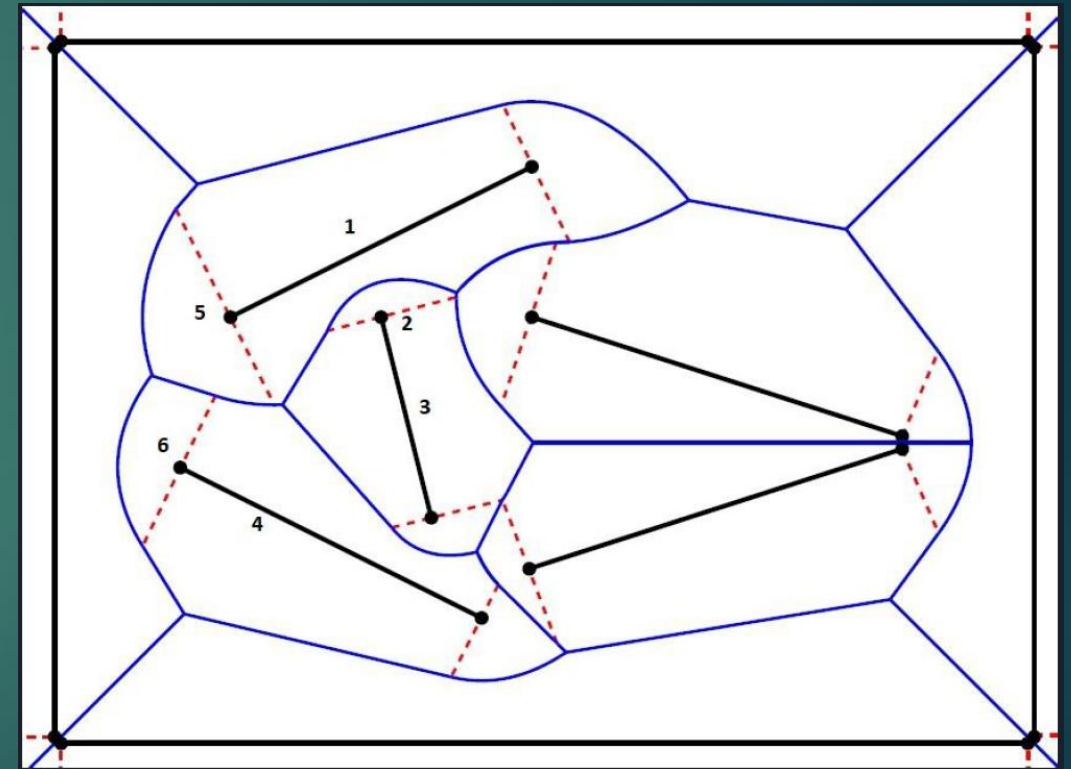


Roadmap

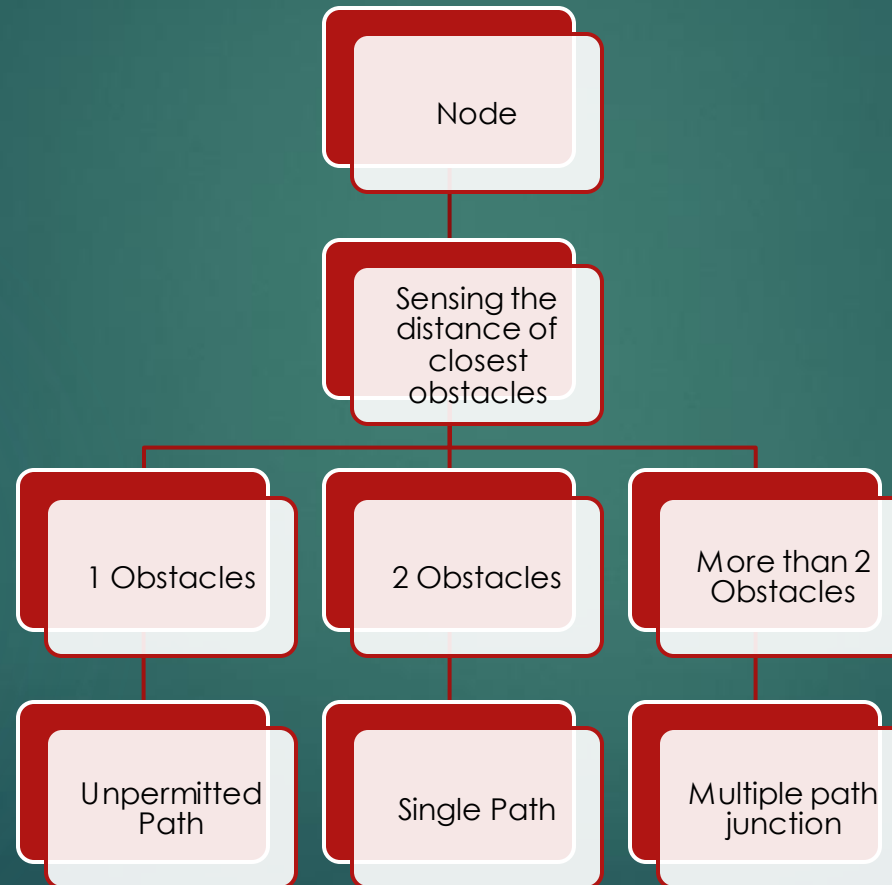
For some general cases, the contours in voronoi diagrams are:

- ▶ Between point – point
- ▶ Between line – point
- ▶ Between line – line

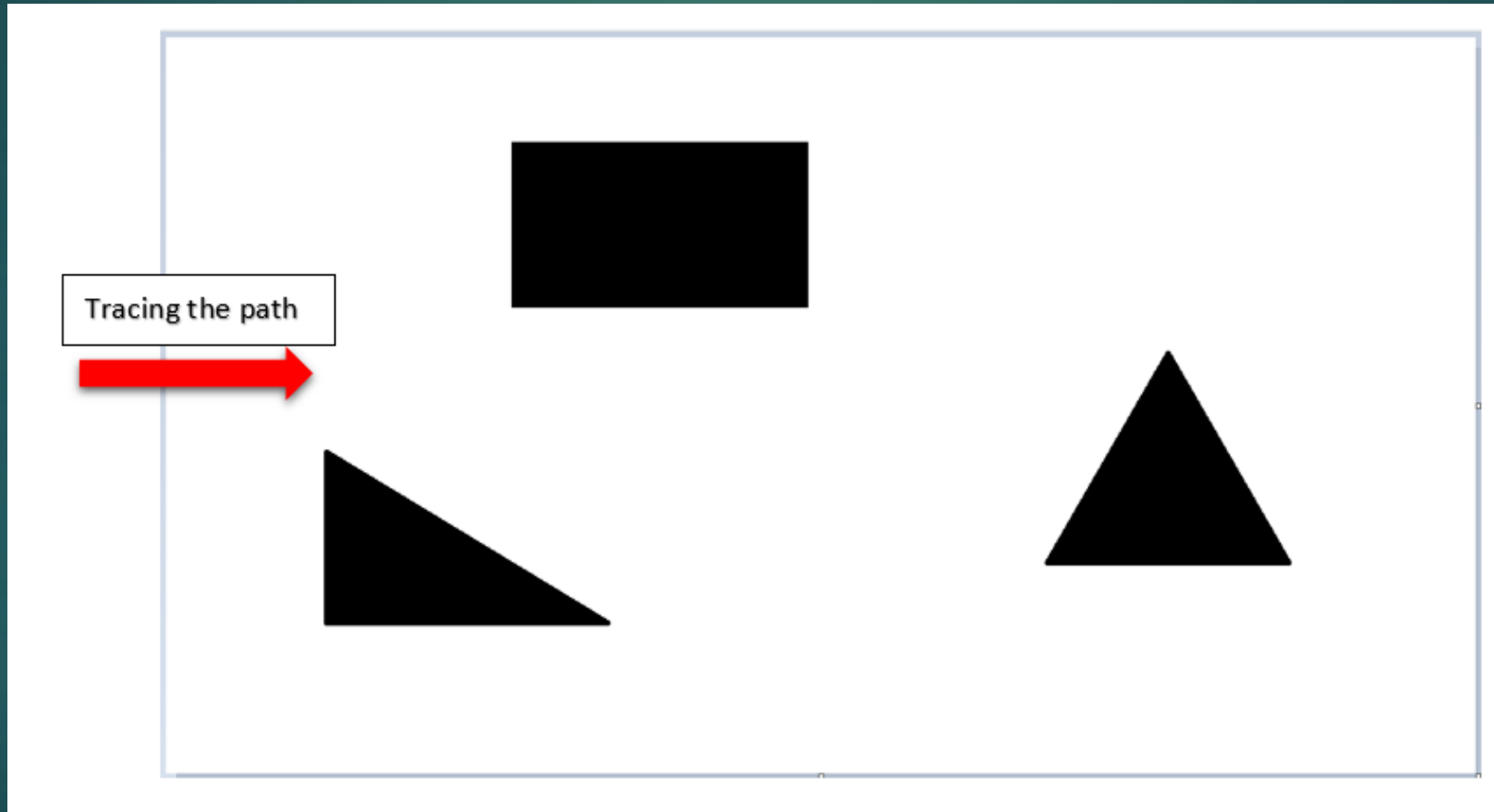
Bisector line
Parabola
Angular
bisector



Basic outline of Algorithm



Implementation

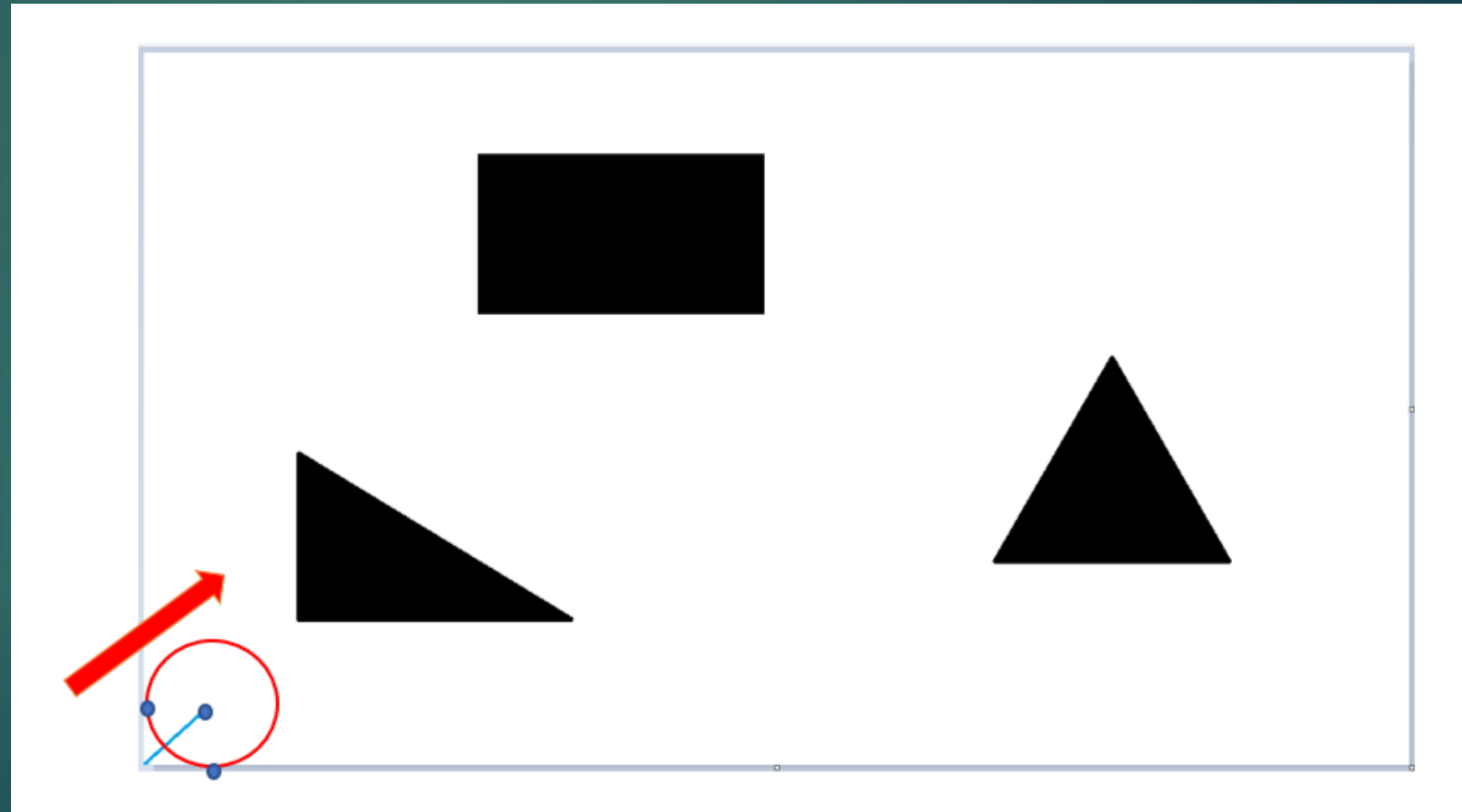


Implementation

Aim – Tracing the least proximity path to the obstacle

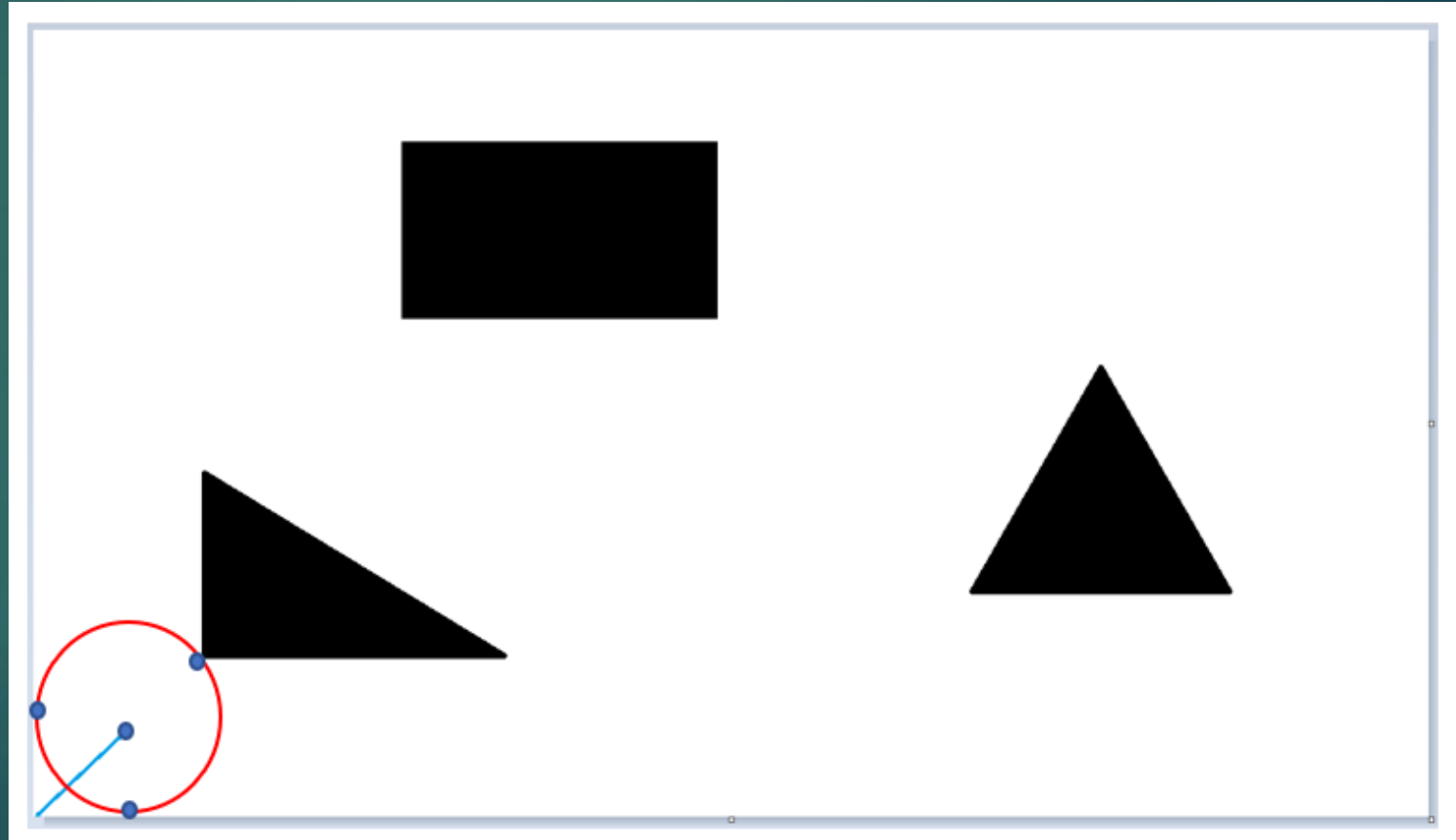
Detecting the two Walls (Obstacles)

Directing (preferring) toward the earlier path direction.



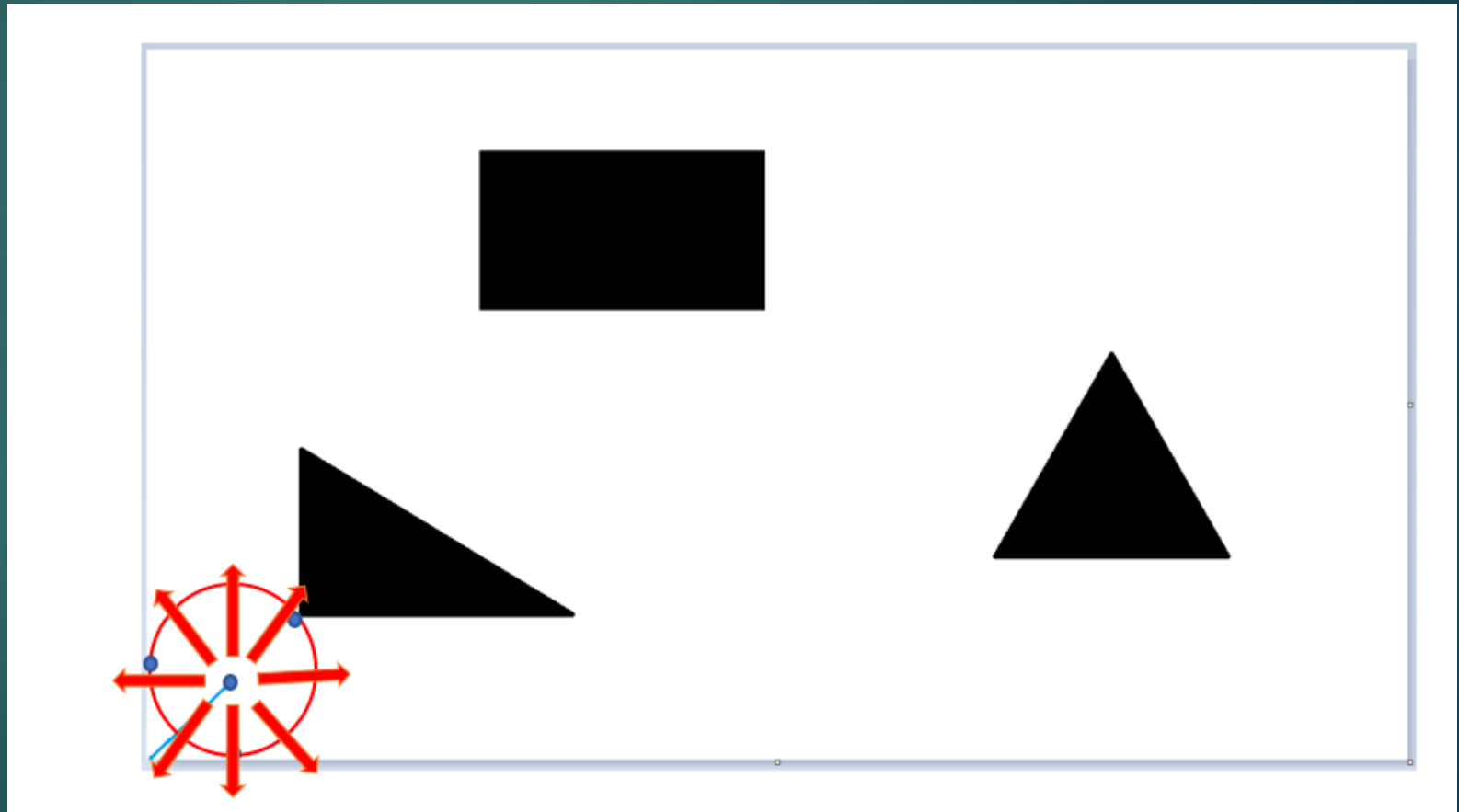
Implementation

Tracing path bifurcate at the point of three (or more) obstacle detection by the sensor.



Implementation

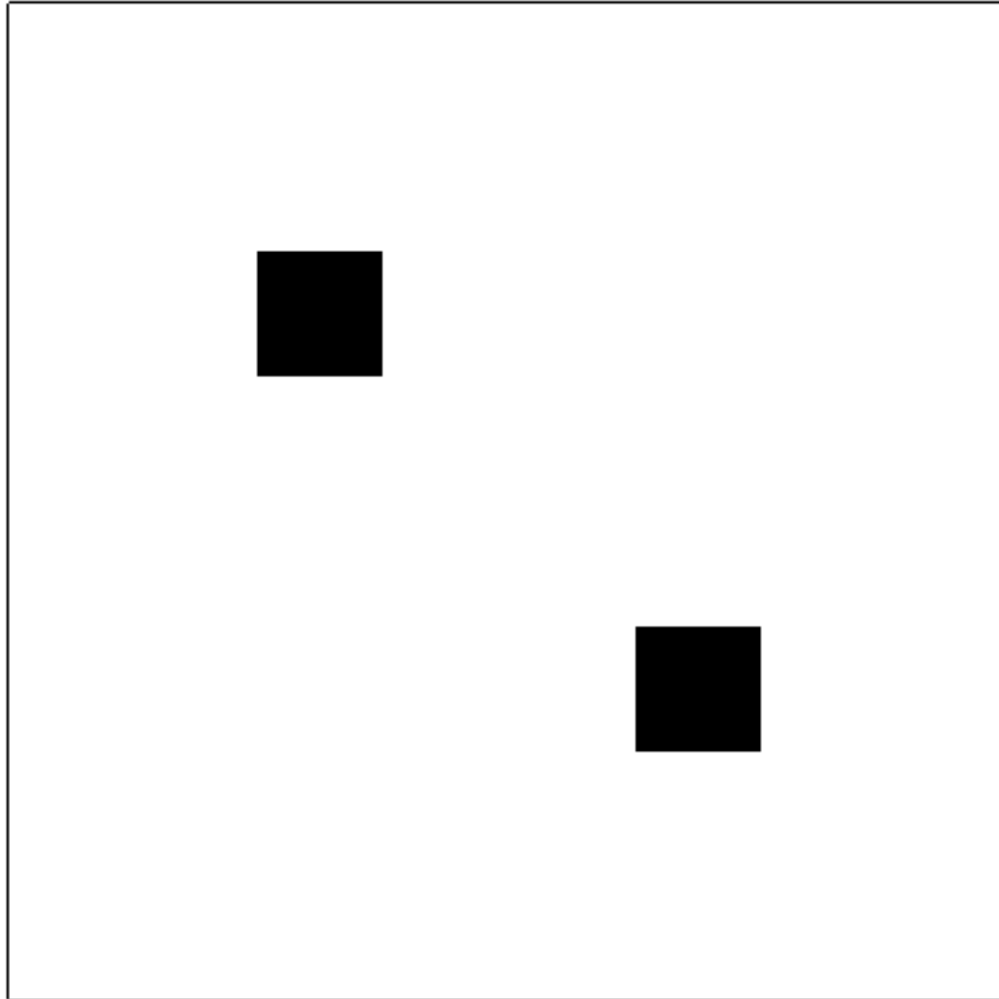
On detecting more than two obstacles, all the eight direction (except the incoming direction) are checked for being the potential path.



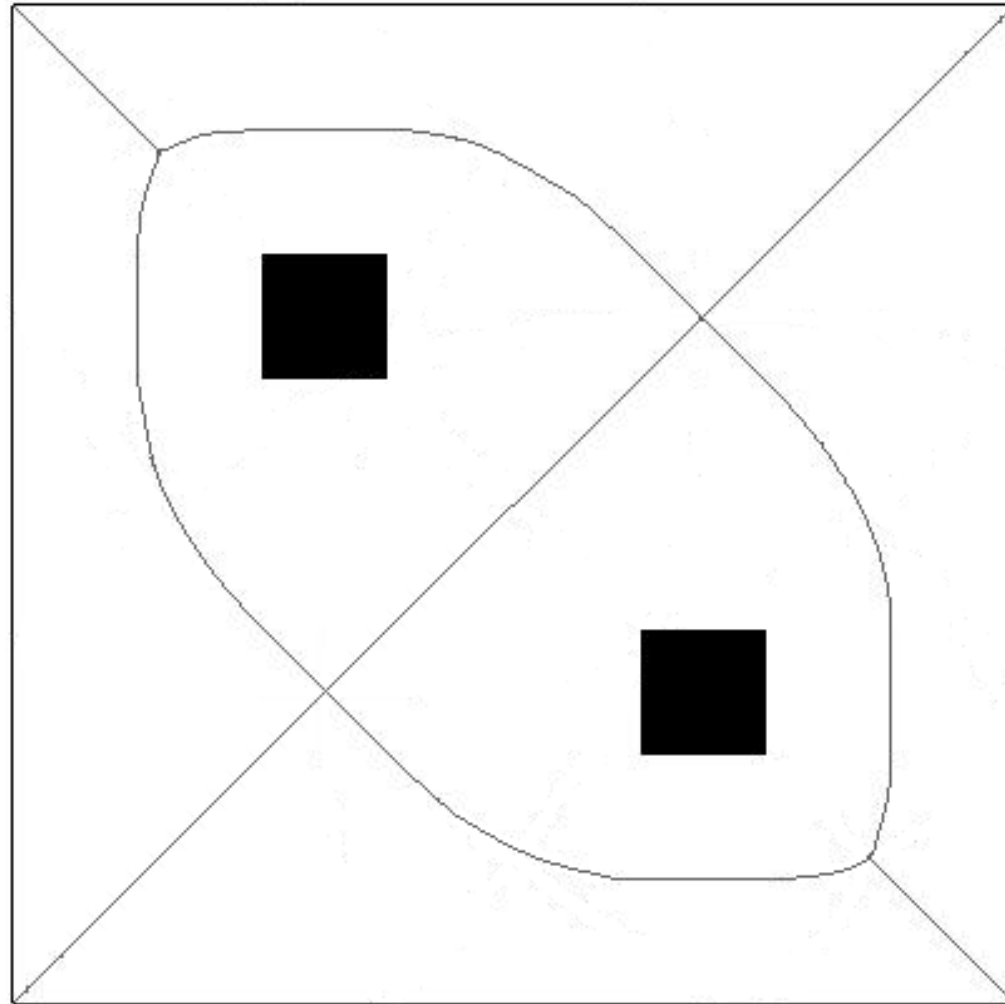
Implementation

- ▶ The resolution of all the images used is 600x600 pixels
- ▶ Initially the robot map image is provided to the program
- ▶ Road map of the image is created
- ▶ The initial position of the robot and the goal position are specified

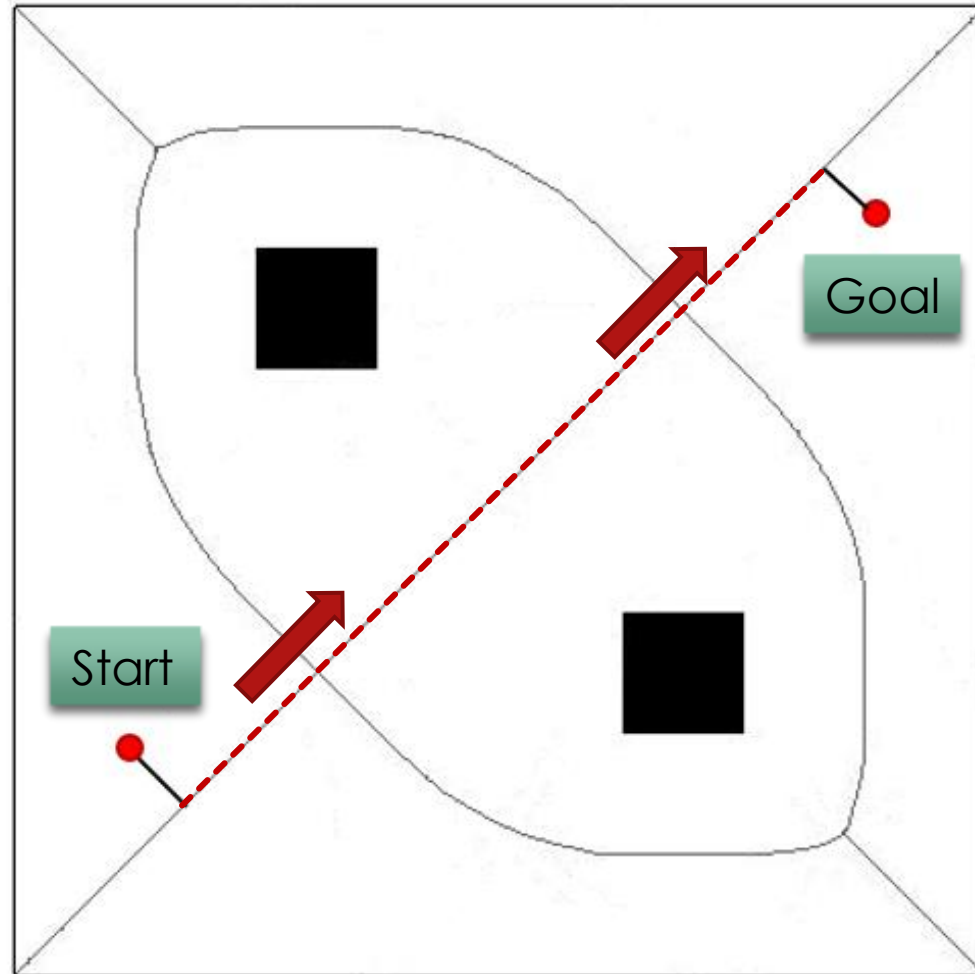
Test Case 1



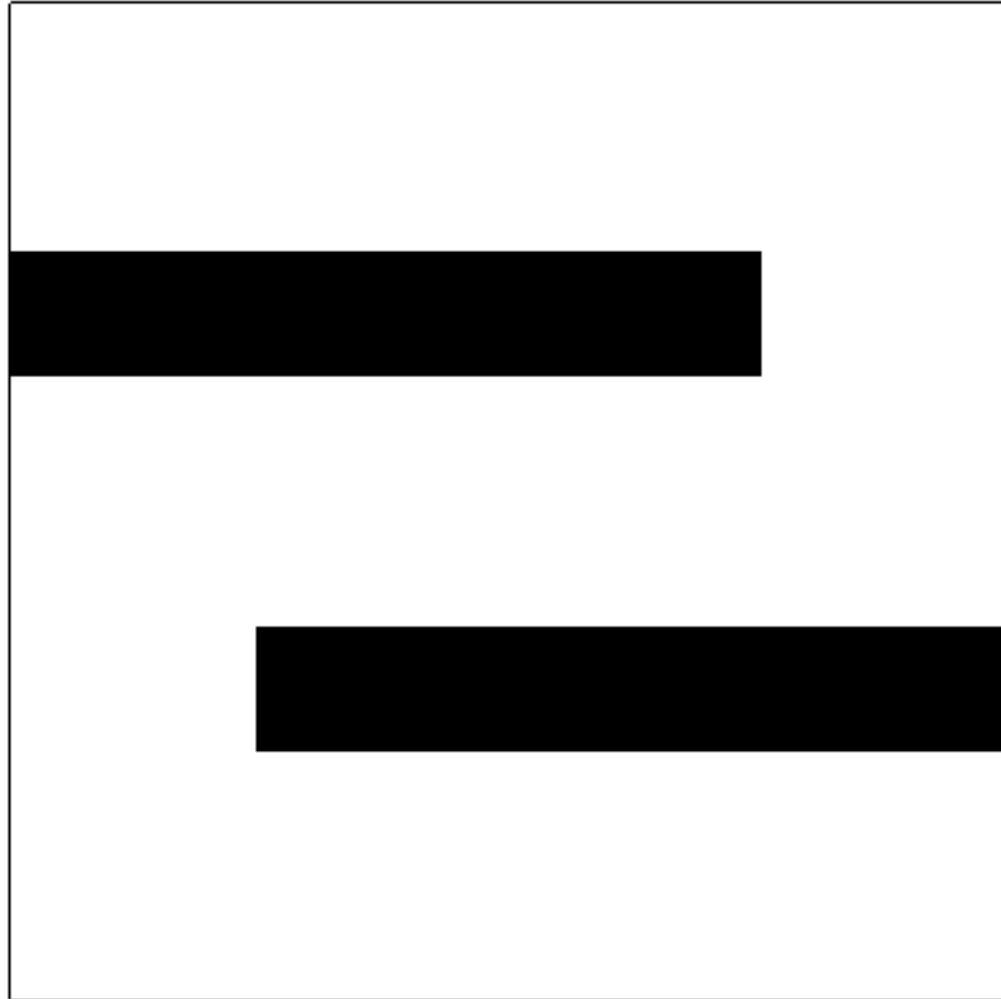
Test Case 1



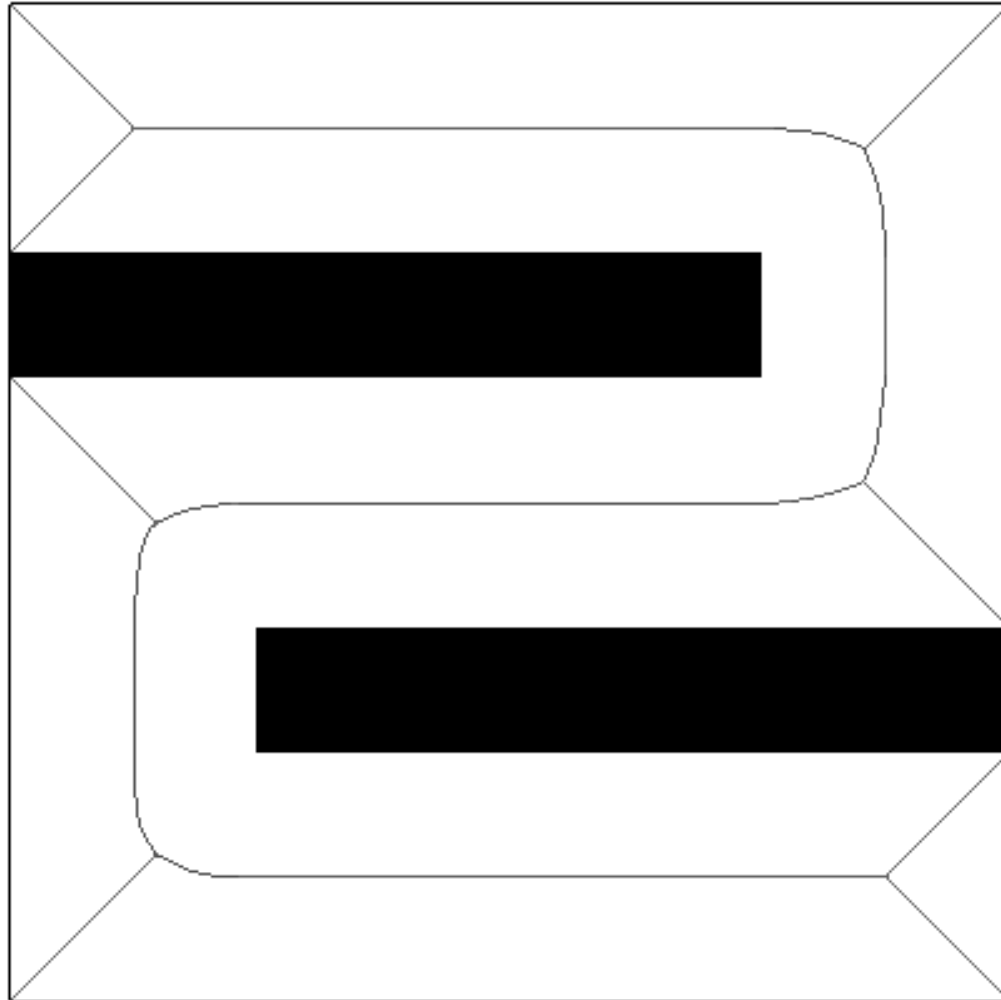
Test Case 1



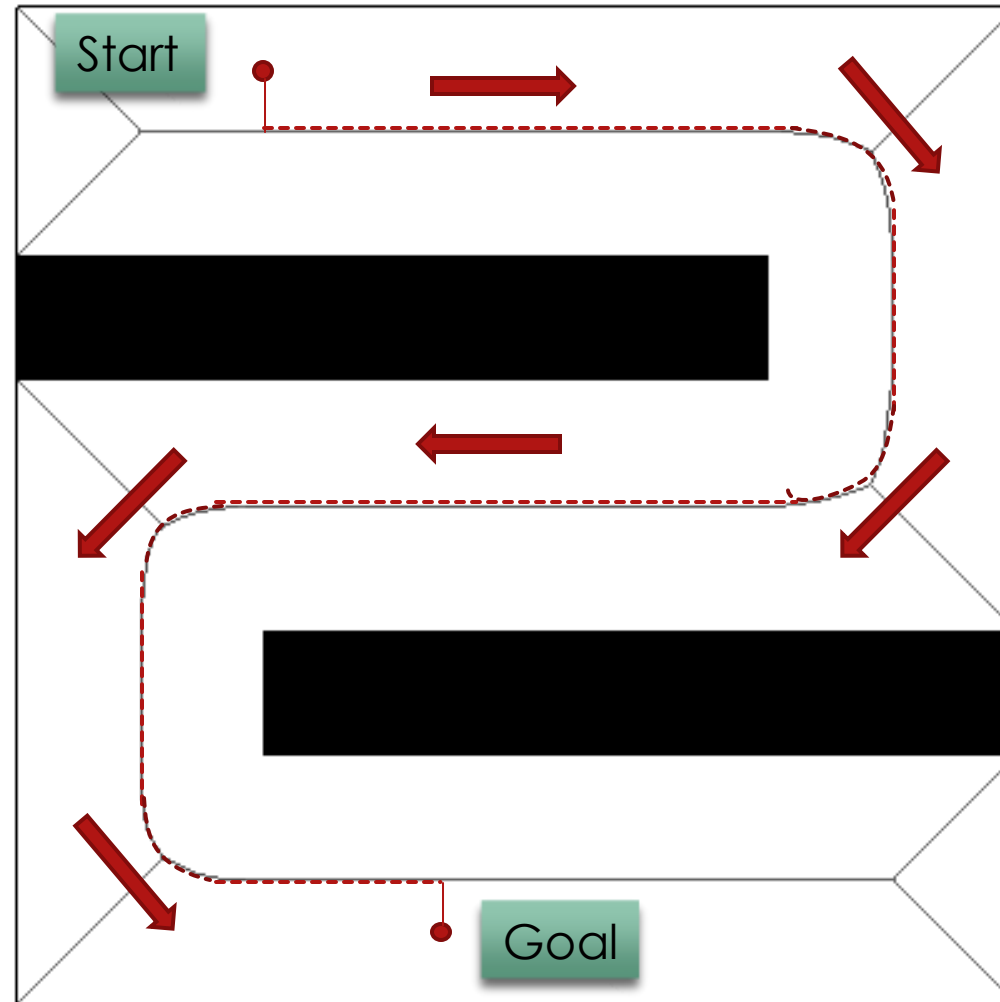
Test Case 2



Test Case 2



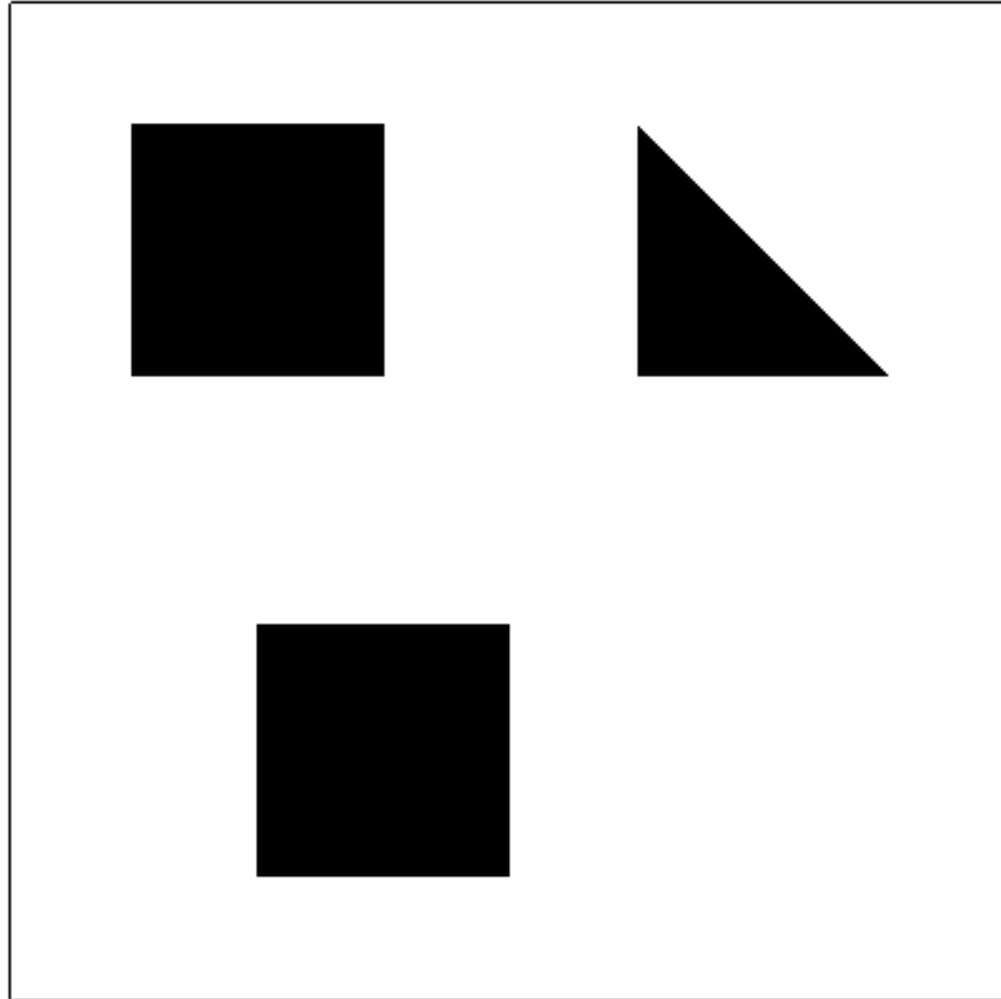
Test Case 2



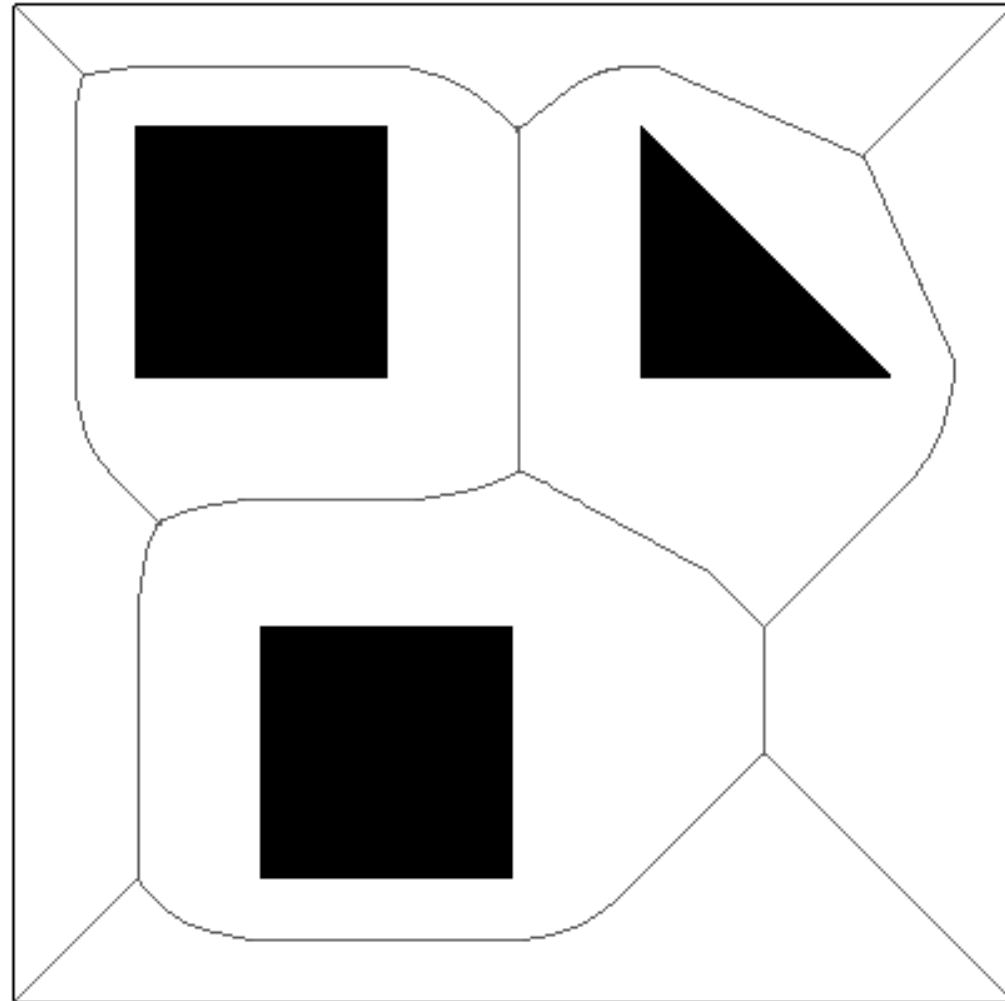


Few more examples

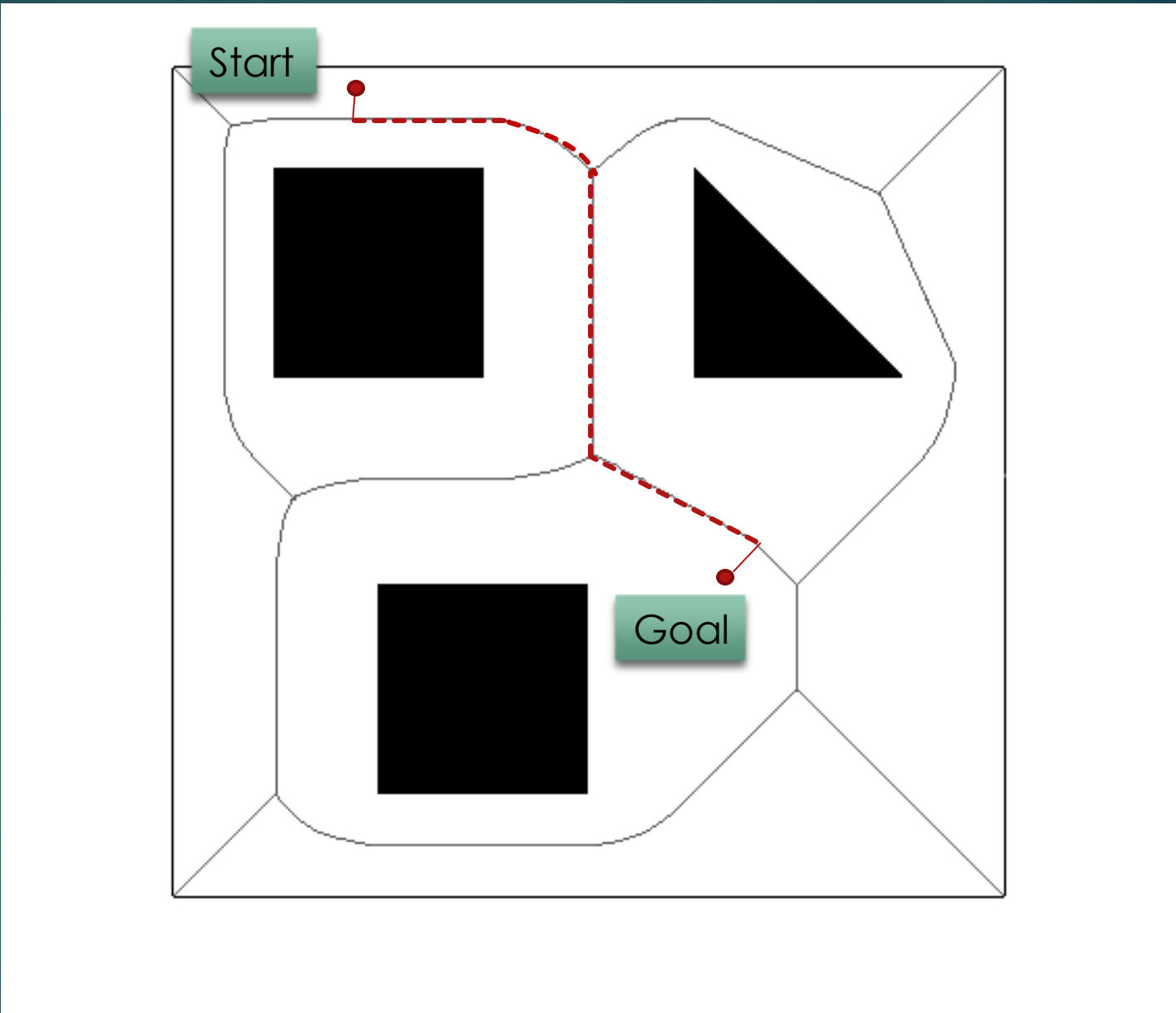
Test Case 3



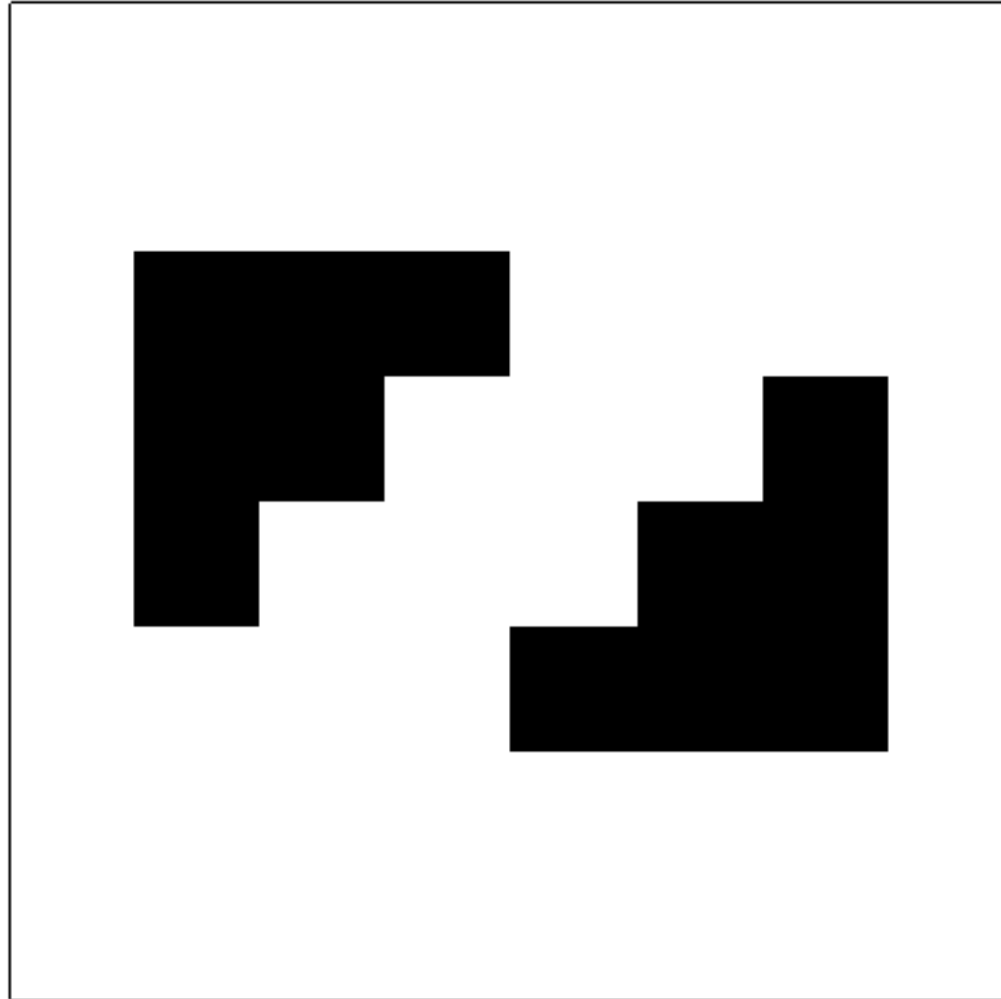
Test Case 3



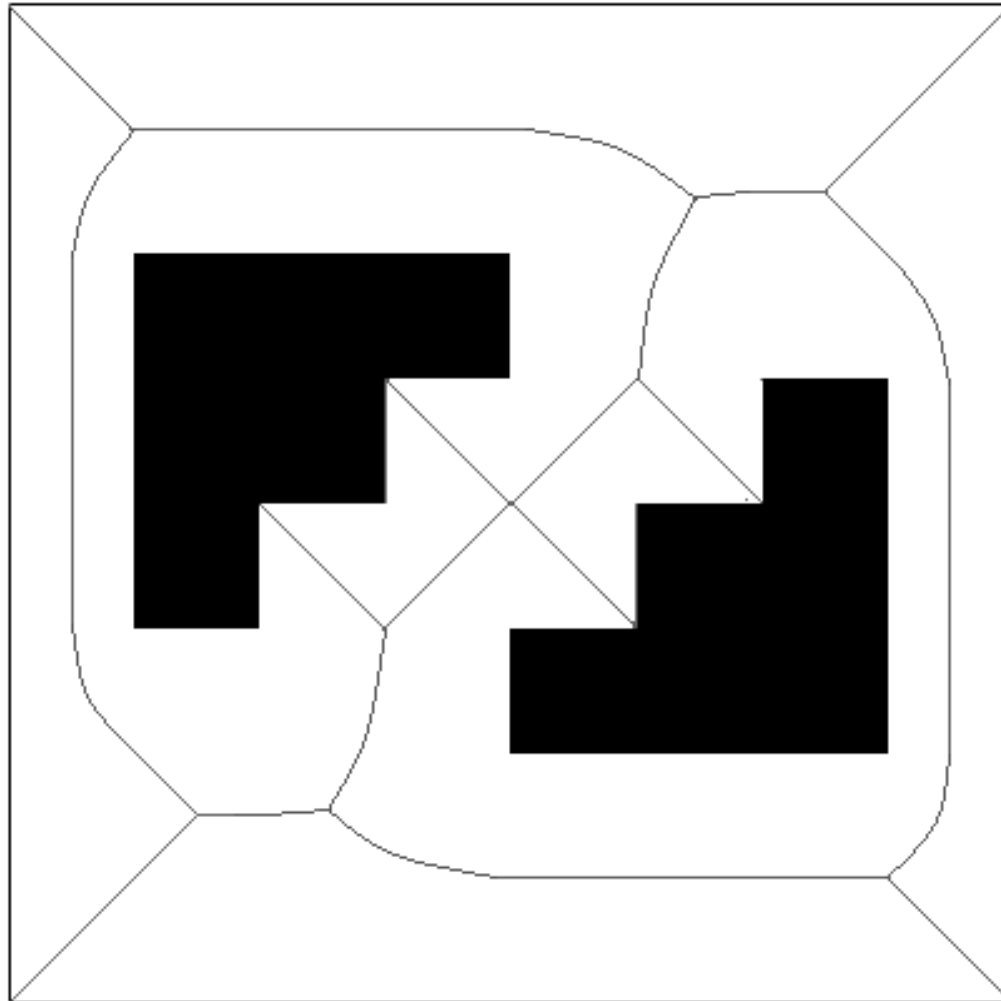
Test Case 3



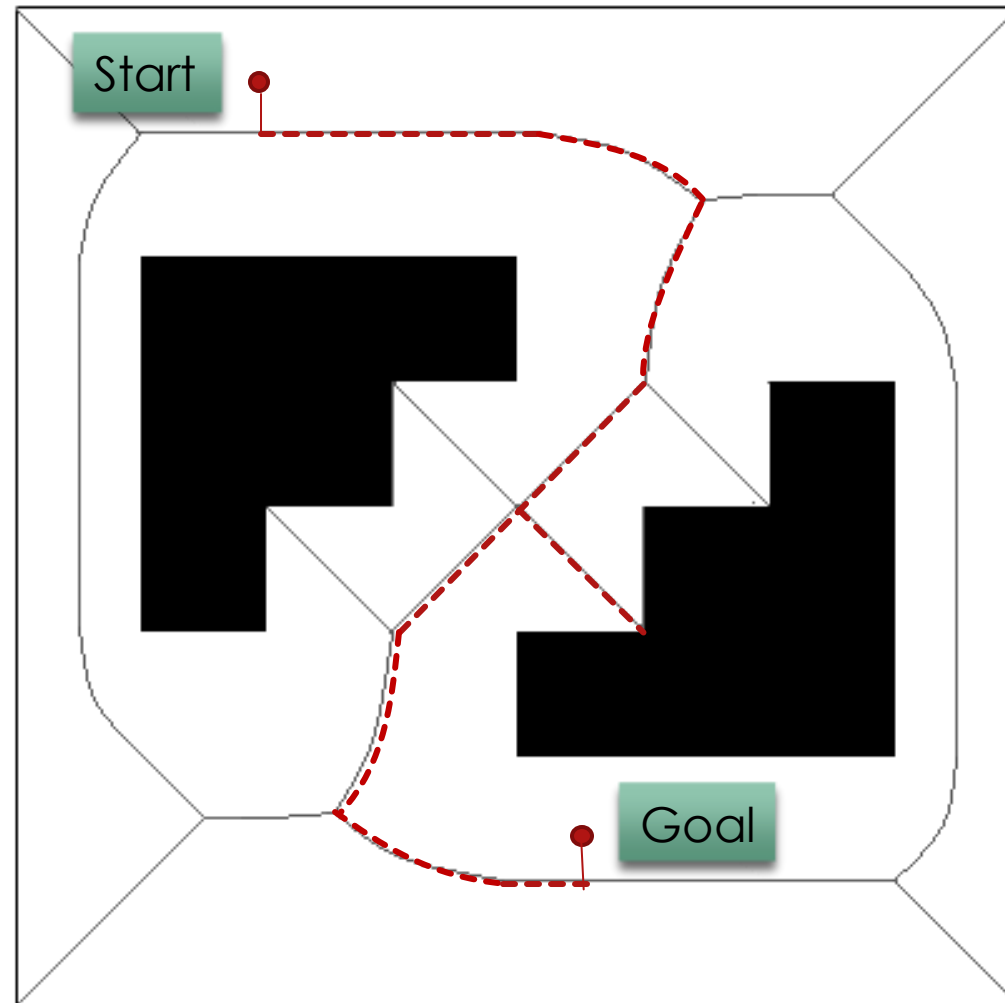
Test Case 4



Test Case 4



Test Case 4



Thank you