

LABYRINTH

Overview

Labyrinth is a Motion Planning Event using ROS and Gazebo. Simulate a real-world problem in which a single agent has to map and navigate through a complex environment using ROS Navigation Stack and classical path planning algorithms with Dynamic Obstacle Avoidance.

So, bring out the explorer in you and relive the age-old sci-fi fantasies of using laser scanners and cameras to dodge and escape undesirable elements in an unknown environment.

Task Description

On the agonizing day of 18 September 2016, four heavily armed militants attacked the brigade headquarters at Uri, Jammu and Kashmir at dawn. Major Vihaan Singh Shergill has been chosen to lead a team to give a befitting reply to the perpetrators of the attack.

An Indian Spy who works for Indian Spy Agency, ISA, has identified a location where the terrorist launch pads are located. While the commandos were undergoing their training, the Major asked Indian Defence Research Institute, IDRI, to assist them in their mission.

You are Ishan, an intern at IDRI, who developed a state-of-the-art robot named Garuda to assist the Major in their mission. But, before deploying it in an actual warzone, you were asked to test out Garuda.

The IDRI imitated a warzone for experimental purpose. To test out Garuda, you have to create an exact map of the imitated warzone.

Then you have to find the shortest path to a given coordinate and make Garuda traverse it. Only after passing these tests Garuda would be commissioned for the actual mission.

After Garuda successfully passed the experiment.,, you were provided with the images of the terrorist launchpad taken by an Indian spy. Now you must extract exact coordinates using the image provided and guide a platoon of the troops to the found coordinate.

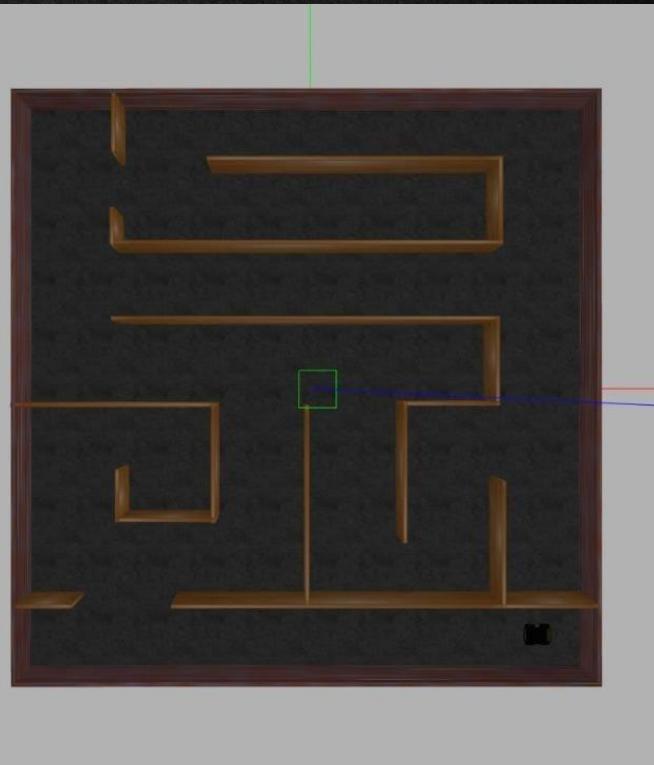
In spite of being outnumbered, the Indian Army killed all the terrorists present at this location, including the ones responsible for Uri Attack. In the enemy territory, the local police were alerted about the possible infiltration by the Indian Army. As a result, the Indian Army was heavily rained down by gunfire from the local police.

Now, help the army exit the enemy territory safely. For this task, find the appropriate exit coordinates using the image provided by spy and then guide the Major and his troops to the exit location where a rescue-chopper is waiting for them.

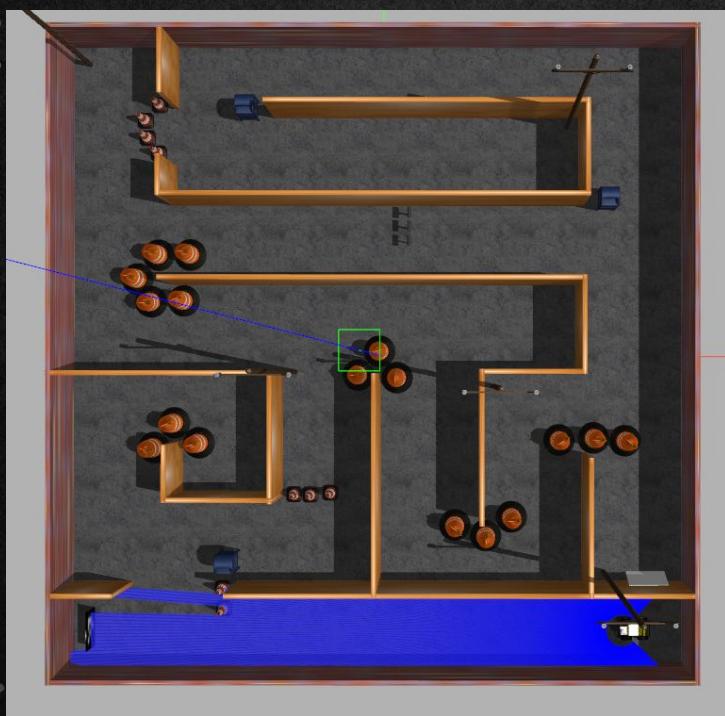
Arena Description

The arena provided in this event is a maze whose images are shown below.

Qualifying Round Arena



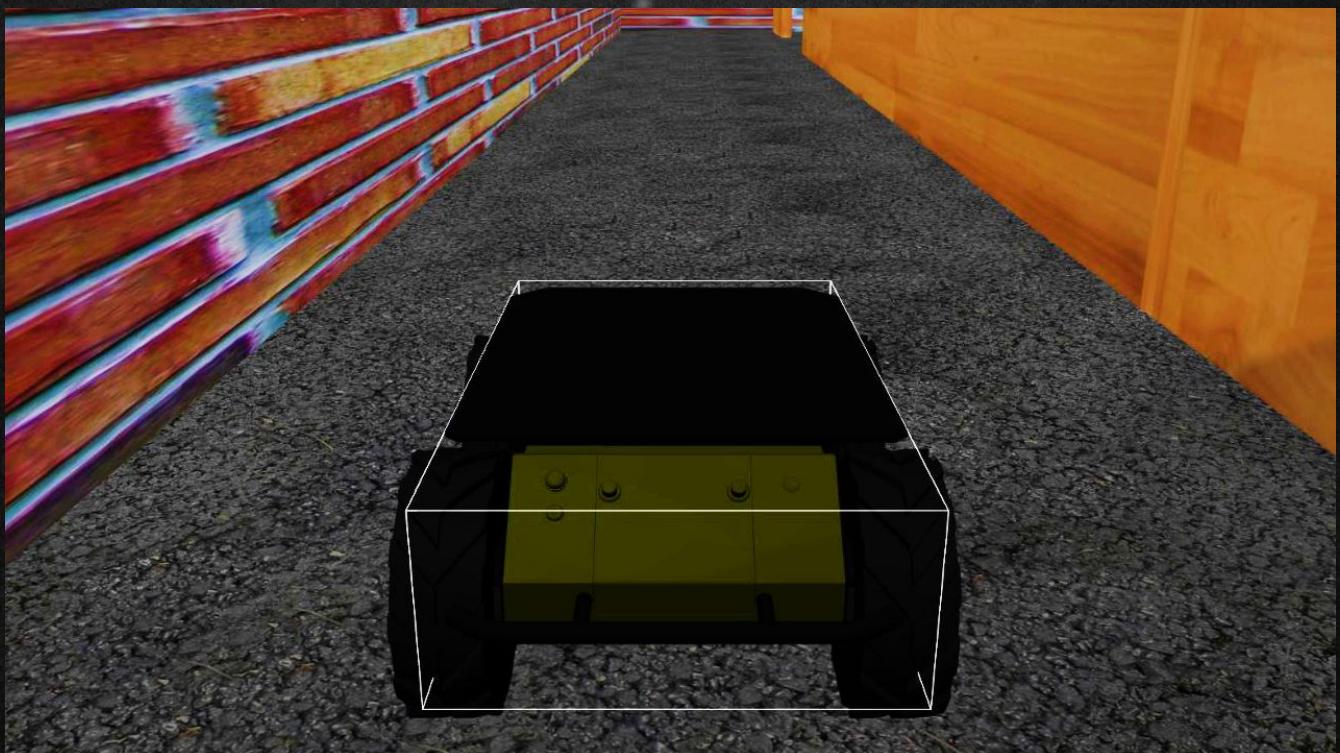
Final Round Arena



Bot Description

In this event, the teams would be provided with the ROS package that contains the simulation environment and a four-wheel robot model. The Robot model has the following sensors:

1. 2D LIDAR
2. Intel Realsense D435i RGBD Camera
3. IMU

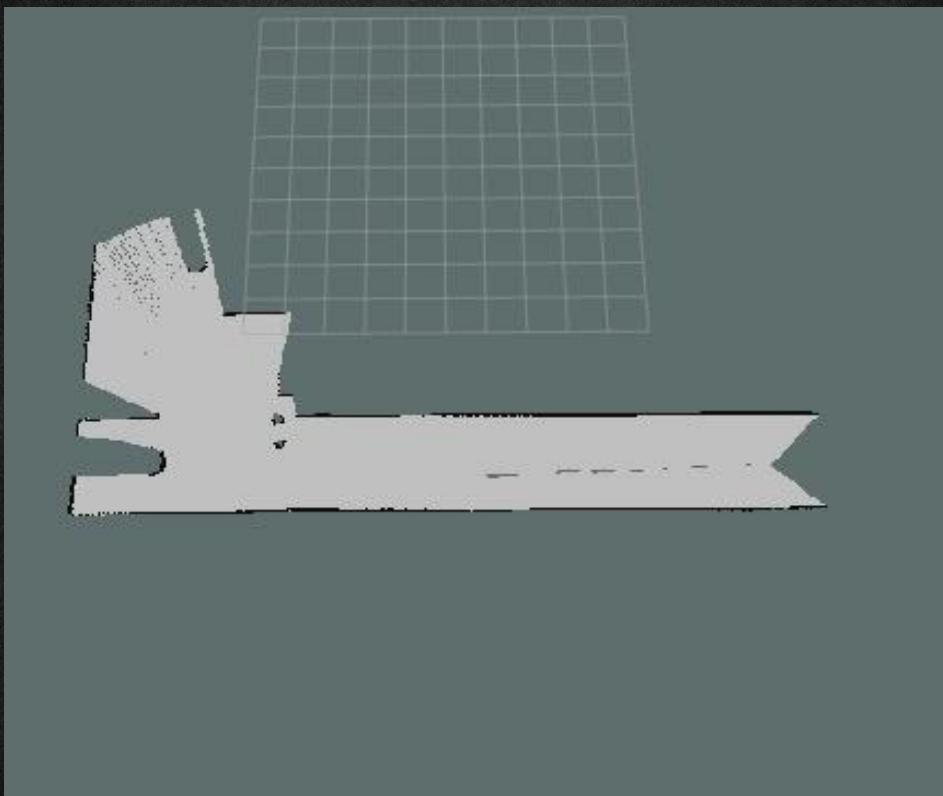


Gameplay

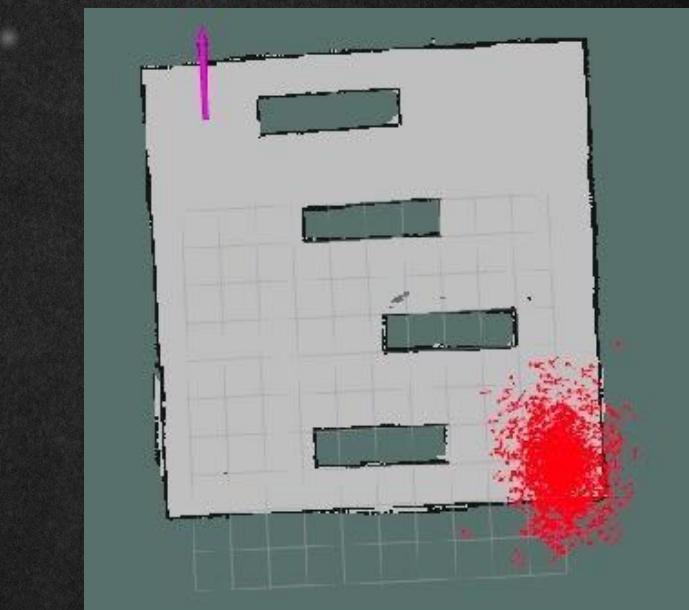
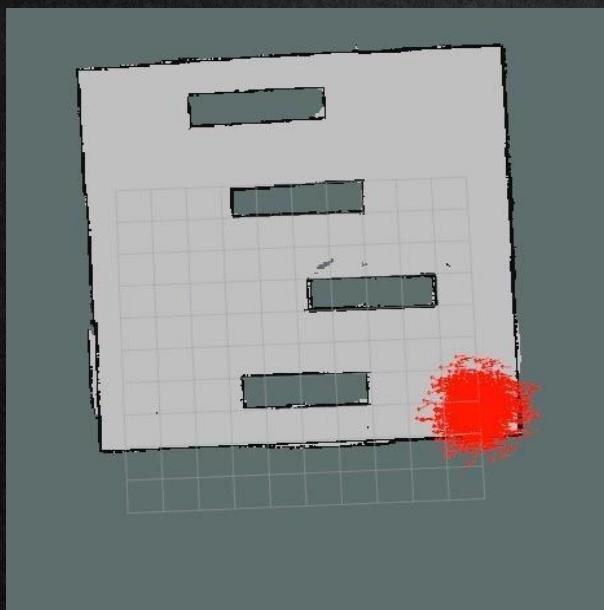
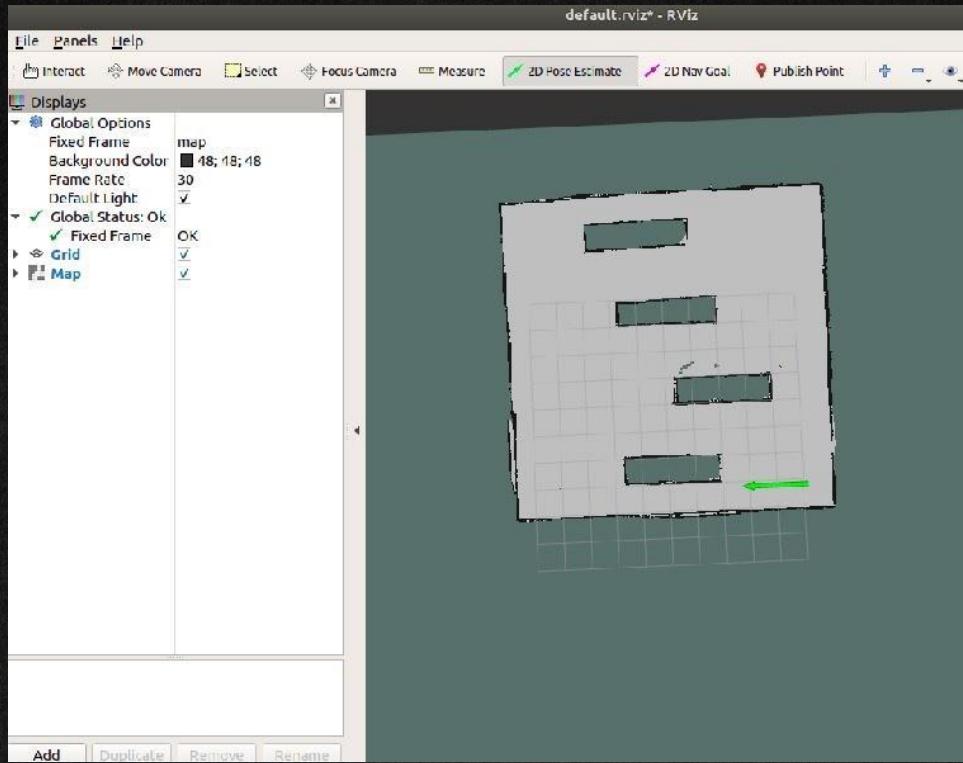
Qualifying Round:

The problem is divided into the following tasks:

1. With the help of Garuda, traverse the entire training arena and create its map using the G-mapping and the Teleop packages of ROS.



2. Now using the map, localize (AMCL) Garuda and help the troops reach the goal using path planning with the help of the ROS navigation stack.



Note: These images are for reference purpose only.

Final Round:

Complete the tasks mentioned in Gaming procedure below.

Gaming Procedure:

- At your initial position, first scan the (given) ArUco Marker using the camera .



- Publish the scanned ArUco marker id to the topic named “/aruco” whose msg type is int8.

- Launch a node provided by us known as the **SPY**. The SPY Node will verify the aruco id and will publish a message in the topic **"/terrorist_location"**.

The message will contain

- 1D array of int8 (flattened image of size 379x1049x3 representing goal position)
- 1D array of float32 (required orientation at goal)
- If it is correct, you will get the image of a terrorist launchpad.
- Using the image provided by SPY, extract the coordinates of the launchpad.

- Simultaneously map the arena and navigate the Garuda to the extracted coordinate.
- Once you reach the correct location, you will find another ArUco Marker.
- Now, scan the new ArUco Marker and publish it's id to the same "/aruco" topic.
- If Id is correct, then SPY will provide you another image. Use this image to extract the exit coordinates.
- Navigate the Garuda to the exit coordinates to finish the round.
- The final round package will be of the same format as Practice Package that will be provided by us on the day of the event.

Practice Package

A ROS package for practice will be given before the actual competition.

Practice ROS package can be found [here](#) 2 weeks before final round.

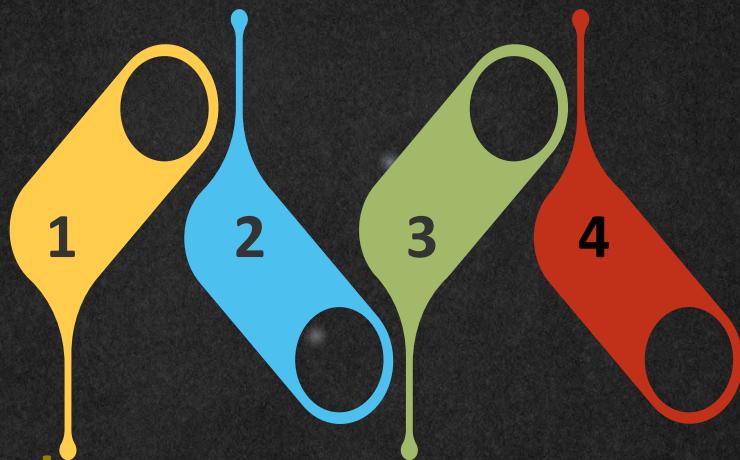
Package Contains

1. Practice Arena
2. Launch files which will spawn the robot in practice arena
3. SPY node that publishes a topic whose msg contains
 - A 1D array of int8 (flattened image of size 379x1049x3 representing goal position)
 - A float32 array which contains orientation of robot
4. SPY node will publish data only if: You have published correct aruco marker Id and you are at correct location as mentioned in game play.
5. You need to subscribe to the topic published to get the information about image and orientation of next coordinates.

You can use this practice arena to make your packages and code work ready.

The below flow chart represents the gameplay of final round described in the previous slide:

**Extract Coordinates
from the given image
and navigate to it**



**Scan ArUco marker
and publish the Id
in given topic**

**Extract Coordinates
from new image
and navigate to it**

**Scan the ArUco marker
at the new location and
publish Id to the given
topic to get new image**

The following images represents the symbols and their values. The position coordinates will be encoded using these symbols which you have to decode using image processing:

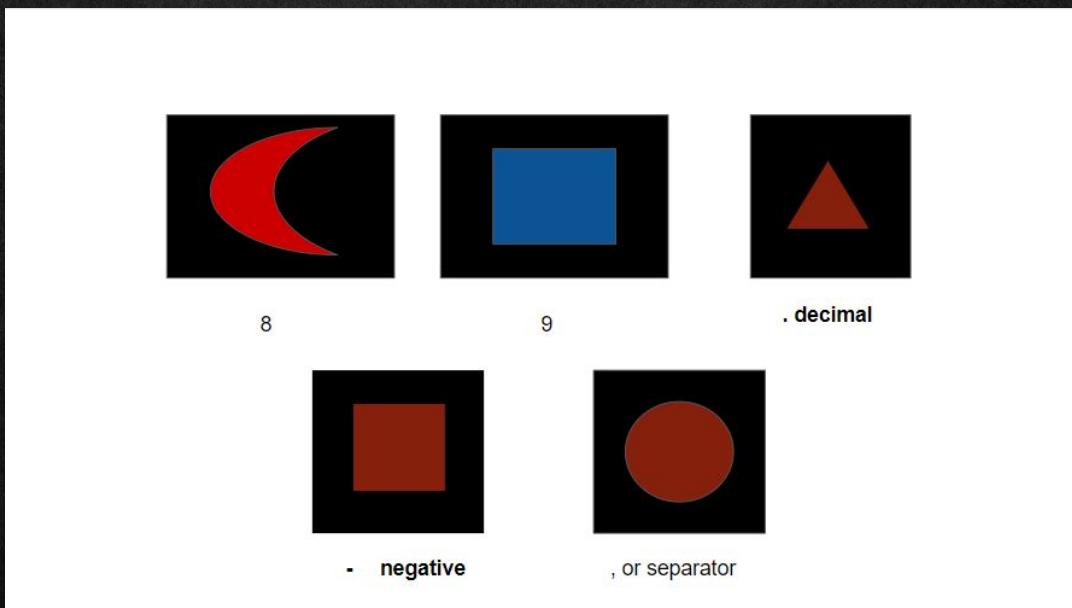
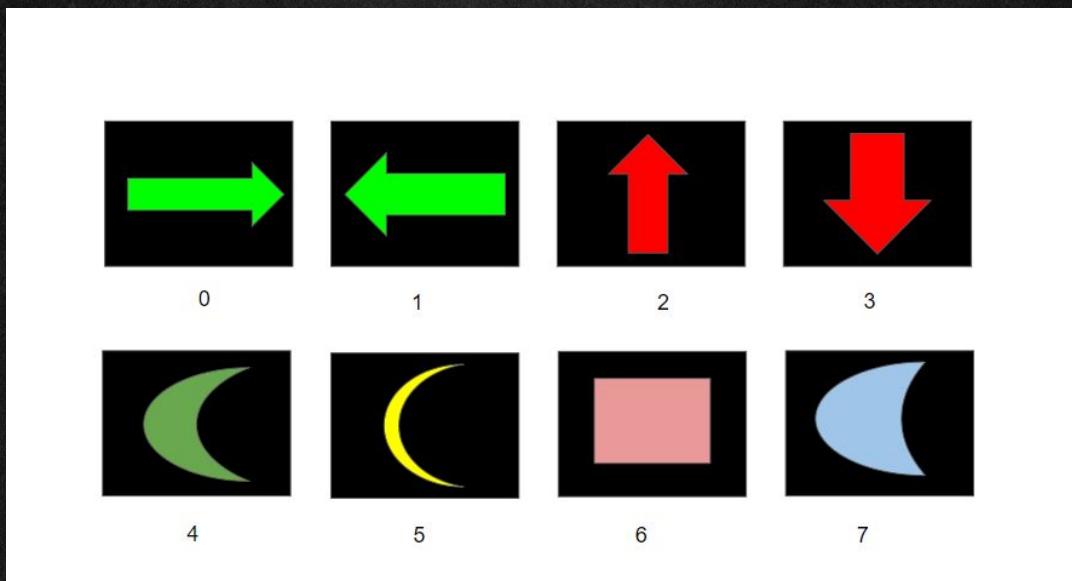
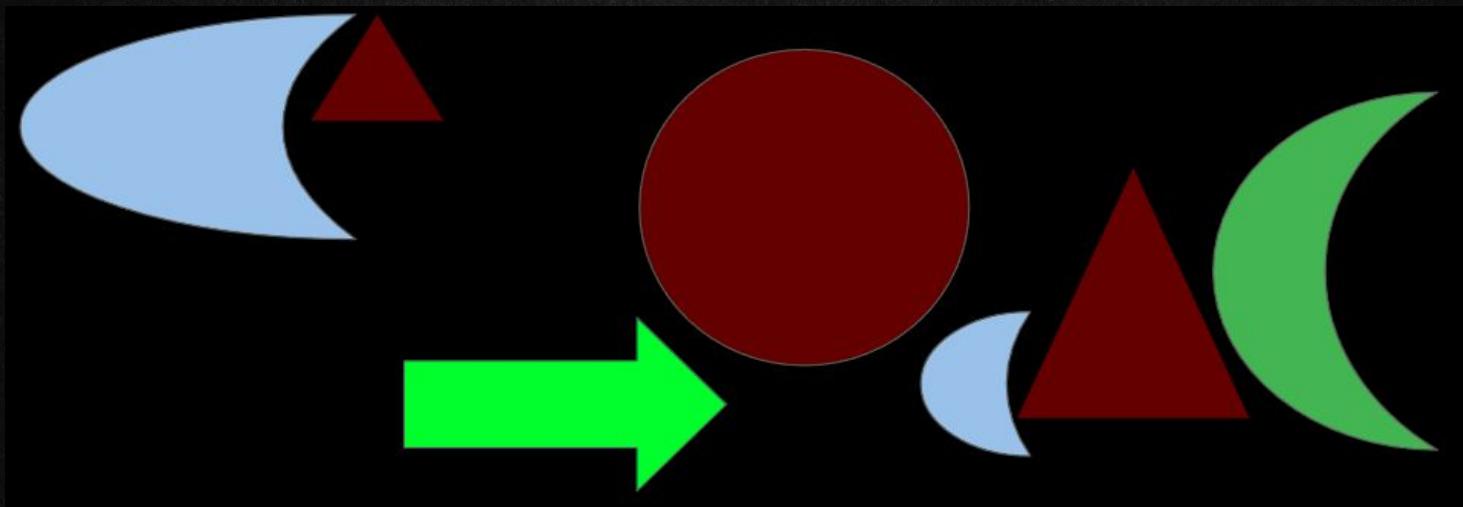


Image Description



An example of how to do image processing is shown below,

The given image represents the coordinate (7.0, 7.4)

Here, you can see that the first symbol represents “7”, as blue is dominant color and shape is moon, the triangle represents a decimal point “.” and the right arrow represents zero. Next the circle is a separator which represents “,”. So the elements following the circle would represent the y coordinate. The first symbol following the circle represents 7, the next triangle represents a decimal point “.” and the rightmost symbol represents 4 as dominant color is green and shape is moon.

Note: All the representations can be found out using the chart shown in the previous slides.

Eligibility

- All students with a valid identity card of their respective educational institutions are eligible to participate in the event.

Teams Specification

- A team may consist of a maximum of 3 members.
- Members of a team can be from different educational institutions.

Event Rules

- Participant should have Ubuntu operating system and ROS installed in system.
- Your evaluation will be according to the rules mentioned in the problem statement.
- Arena for qualifying round will be released a day before the beginning of the event.
- Arena for final round will be released on the day of submission.
- The final evaluation will be on the ROS platform.

- You need to choose your algorithm according to the restrictions mentioned in the problem statement.
- The organizers reserve the right to change the rules as they deem fit. Change in rules, if any, will be highlighted on the website and notified to the registered participants.
- The decision of the organizers shall be final and binding.

Submission procedure

- Qualifying round arena will be available [here](#) 1 day before the submission date. The goal position and orientation will be provided in a txt file.
- For qualifying round, you are required to submit a solution video (less than 2 minutes) along with the ROS package. The video must contain terminal window.
- Name your file as,
`team name _ college name _ labyrinth`
- Video must be in mp4 format.
- ROS package must be zip format.
- Submit your code and video through this [link](#).

- For final round, a meeting link will be provided, where you will present your screen. Node and arena will be given to you on the spot and you have to run your code.

Scoring

- The time taken (in seconds) by you to complete the task will be converted in points as;
$$500 - (\text{time taken})$$

For example,

If you took 2 minutes (120 seconds) to complete the task, then your points will be, $500 - 120 = 380$.

- There are no points for 1st round. It is a qualifying round.
- Each collision with the obstacle will cost you a penalty of 10 points.

Certificate Policy

- The top three teams will be awarded a certificate of excellence.
- All the teams qualifying the first round will be awarded a certificate of participation.
- Disqualified teams will not be considered for any certificates.

Timeline

- Arena for Qualifying Round will be provided on 8th March.
- Deadline for Qualifying round solution submission is 9th March 11:59 PM.
- A practice arena will be provided 2 weeks before the final round.
- Final round will be held on 13th March.
- Meeting link for final round will be provided to the qualifying candidates via email.

Contact:

Mainak Samanta: 7872136912

Bhuvnesh Soni: 9462258170