Thank you for participating in eYRC 2020-21. Hope you found this learning experience useful.
We regret to inform you that your team is **NOT shortlisted as a finalist in eYRC 2020-21**. We value your interest in e-Yantra and hope we have a chance to engage with you in the near future.
Wishing you all the very best in your future endeavors.
-- **e-Yantra Team**

Eligible for Certificate Type: **Certificate of Completion**
We are updating the Certificate Status.. Above status might get changed.

DOWNLOAD CERTIFICATE

Task 6 Result:

Task	Marks	Penalty	Total Marks
Original Configuration	2408.90	0.00	2408.9
Bonus Configuration	765.67	0	765.67
Code	80	0	80

Remark: For Demonstration Video:
Well done. Algorithm should be optimized to increase its speed and efficiency. Try to determine and eliminate the corner cases in your code. Improvisation in the control logic is required to reduce the Hit Penalties. This will help to achieve a better score.

For Code:
Code is very well written, divided into separate modules making it easy for one to understand and read the logic. Good strategy to use threads in implementing the solution, but instead of having one thread for each ball, you could have a thread for 4 tables, as the number of balls to traverse can vary but not the number of tables in the scene. But, overall good efforts shown. Also you made use of Jupyter Notebook to implement parts of your logic and then created py file out of it. Keep it up.

From Instructors:
Heartiest congratulations to the team for putting in efforts to reach till here. We appreciate your work and hoping that you have learnt a lot and enjoyed at the same time. We hope to see you back with more energy and enthusiasm. Congratulations and All the Best!

Task 5 Result:

Marks	Penalty	Total Marks
81.00	0.00	81

Remark: Marks scored for Control logic and Navigation (out of 80): 63

Remarks for Control logic and Navigation: Good job. 'task_5_output.txt' evaluated successfully. The least time to complete traversal on Table 4 was 10.95 simulation seconds. The least time to complete traversal on Table 1 was 8.95 simulation seconds. Optimize your algorithm further. As mentioned in the 'Judging and Scoring' section of the Rulebook, each collision will result in a penalty of 10 marks. Tune your control logic further. Note: Marks calculated in this task are with respect to the top scoring team.

Marks scored for Video (out of 20): 18

Remarks for Video: Keep up the good work. Read the 'General Instructions' mentioned in the 'App for Grading of Progress Task' documentation. Video recorder initialization was not followed.

TASK 5 GRAPH

Task 4 Result:

Task	Marks	Penalty	Total Marks
Task 4A	20.00	0.00	20
Task 4B	35.00	0.00	35
Task 4C	17.00	0.00	17

Remark: For Task 4A: Way to go!! The generated output.csv cleared all the testcases

For Task 4B:

Marks scored for Control logic and Navigation (out of 54): 35

Remarks for Control logic and Navigation: Good job. Your 'task_4b_output.txt' was evaluated successfully. Good job on traversing the entire path successfully.

The least time to complete entire Task 4B was 44.35 simulation seconds. Work hard to eliminate the bottlenecks in your code to increase its speed. Improvise your algorithm/design by implementing the methods mentioned in documentation and piazza posts.

The necessary respondable elements were not found.

Good job. Your 'task_4b_output.xml' was evaluated successfully.

However, the following points led to deduction of marks.

Incorrect numbering convention(s) was/were followed for spherical_joint_ic_4 etc.

Upper case was/were detected for Revolute_joint_cc_1 etc.

Marks scored for Video (out of 6): 6

Remarks for Video: Well done. Try to reduce the time taken to complete the task. Follow the naming convention as mentioned in the documentation. A penalty of 6 marks has been imposed for not following the same.

For Task 4C:

Q2) Correct Answer

Q3) Correct Answer

Q4) Correct Images given

Q5) Correct score calculated

Q8) Good explanation given for path planning

Q9) You have not elaborated much about algorithm implementation. You need to think about this a bit more

TASK 4 GRAPH

Task 3 Result:

Marks	Penalty	Total Marks
81	0	81

Remark: **Marks scored for Control logic and Design (out of 90):** 77

Remarks for Control logic and Design: Good job. Your 'task_3_output.txt' was evaluated successfully.
However, the following points led to deduction of marks.
Improving the responsiveness of your control logic is necessary.
The necessary respondable elements were not found.
Good job. Your 'task_3_output.xml' was evaluated successfully.
However, the following points led to deduction of marks.
Incorrect numbering convention(s) was/were followed for spherical_joint_ic_4 etc.
Incorrect numbering convention(s) was/were followed for base_plate_respondable etc.
Incorrect 1st initial was/were found for spherical_joint_ic_2 etc.
Incorrect 2nd initial was/were found for spherical_joint_ic_2 etc.
Upper case was/were detected for Revolute_joint_cc_1 etc.

Marks scored for Video (out of 10): 9

Remarks for Video: Well done. Try grouping different objects in the scene together to simplify the hierarchy. Use correct names as mentioned in the task 1c documentation.

TASK 3 GRAPH

Task	Marks	Penalty	Total Marks
Task 2A	13.00	0.00	13
Task 2B	70.00	0.00	70

Remark: **For Task 2A:** Good job, task_2a_output.txt file was evaluated correctly. Your code in task_2a.py and task_1a_part1.py was able to pass very few the test cases successfully, but it failed for most of them. You need to re-work on the scan_image() functionto get correct values of color and centroid for all circles may it be blue, green or red. Keep it up! Your code ran well for test cases having 2 balls but failed for test cases having more than 2 balls and also detecting their color.
For Task 2B: Way to go!! The generated output.csv cleared all the testcases

TASK 2 GRAPH

Task 1 Result:

Task	Marks	Penalty	Total Marks
Task 1A	20.00	5.00	15
Task 1B	30.00	0.00	30
Task 1C	45.00	0.00	45

Remark: **For Task 1A part1:** The area detection wasn't very accurate in the output.txt Apart from this, the output.txt had correct details. The scan_image function passed most of the test cases. Well done!!
For Task 1A part2: Way to go!! The output.txt was perfect. Kudos! The process_video slayed all the testcases .
For Task 1B: Good job, task_1b_output.csv file was evaluated correctly. Good work. Your code in task_1b.py was able to pass most of the test cases successfully, but it failed for few. You can fine tune the applyPerapplyPerspectiveTransform() and detectMaze() functions to get correct encoded values.
For Task 1C XML file: Good job. Your 'task_1c_output.xml' was evaluated successfully.
However, the following points led to deduction of marks.
Incorrect hierarchy for top_plate_respondable
Incorrect 2nd initial was/were found for revolute_joint_ci_2 etc.
WARNING: Upper case was/were detected for Force_sensor_bc_1 etc.
For Task 1C Video file: Use revolute, spherical or prismatic joints where rigid links are not required. For eg, use spherical joint between connecting rod and 'L' connector. Use force sensors where rigid links are required. For eg, use force sensor between 'L' connector and top plate. Group similar objects to simplify the hierarchy. Make sure that two objects do not overlap with each other.

Task 0 Result:

Marks: 86

Penalty: 0.00

Total Marks: 86

Remark: **For task0_output.txt**: Good work. Task 0 output is as expected.
For CodeChef contest: Good performance in the contest. However, we found that 48% of teams failed to use recursion in the D2BIN_PY problem of the CodeChef contest. We attribute this to one of the two reasons - either students failed to read the problem description properly or didn't know how to use recursion. We have penalised such teams by 80% of the score secured in D2BIN_PY problem. To avoid getting penalty in future tasks, kindly read the instructions properly. All the best!

TASK 0 GRAPH