










ERF Challenges & Hackathon

Robotics competitions provide a valuable platform for evaluating the accomplishments of worldwide teams in shared challenge scenarios. They foster discussions, facilitate knowledge exchange, and enhance research.

In the upcoming ERF 2024, participants can engage in 4 thrilling robotics Challenges and 1 Hackathon, namely:

Challenges	
1. Robotic Dog Race	2. Robotics in Agriculture
3. Robotic Manipulation in Manufacturing	4. Drone & Autonomous Vehicle Challenge
Hackathon	
Robot navigation in an urban environment	

Industrial sponsors & Call for Participants

ABB	Power and productivity for a better world™ 
Bonfiglioli	
EagleProjects	
Leonardo	
QBRobotics	
Unitec	
Universal Robots	

Enrolment

Enrolments will be accepted until a maximum number of teams (~ 10) for each challenge will be reached. All interested teams can register for the competition at the following [link](#). General requests shall be addressed to **Prof. Giovanni Berselli** (giovanni.berselli@unige.it)



Prizes

The winning team of each competition will receive a prize of **1000 Euros**. the Robotic Dog Race, the prize will be a **Unitree Go1 Edu Robot**.

As for

Organizers and Points of Contacts



Prof. Giovanni Berselli
 Challenges Supervisor
 General POC
giovanni.berselli@unige.it



Prof. Michele Focchi
 Robotic Dog Race
 POC
michele.focchi@unitn.it



Dr. Dario Mengoli
 Robotics in Agriculture
 POC
dario.mengoli2@unibo.it



Prof. Andrea M. Zanchettin
 Robotic Manipulation in
 Manufacturing POC
andreamaria.zanchettin@polimi.it



Dr. Nicola Mimo
 Robotic Drone Contest &
 Student HackatonPOC
nicola.mimmo2@unibo.it



Dr. Andrea Pupa
 GitHub Repository
 POC
andrea.pupa@unimore.it



1 Robotic Dog Race



EAGLEPROJECTS
 Technology Factory

The Robotic Dog Race Challenge (RDRC) is a competition supported by the company **EagleProjects**, that

will provide a series of **Unitree Go1Edu** legged robots. Following the ICRA 2023 Challenge, the RDRC will add also the autonomous exploration of an unknown environment. The tasks are crafted to engage/test fundamental abilities essential for robotics engineers. Participants will be challenged in problem-solving skills, strategic planning, and creativity to complete the task.

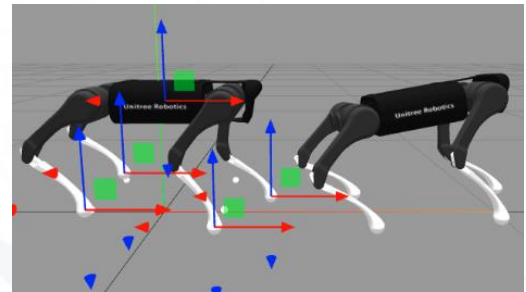


Description

The Dog challenge is divided in two tracks:

1. **Locomotion Track:** the robot should race from point A to point B in the shortest time overcoming different obstacles (ramps, debris), possibly minimizing the collisions of the body with the environment; the robot can be teleoperated.
2. **Navigation Track:** the robot should explore autonomously an unknown environment (e.g. a maze) to find a given object.

Teams will be provided with an open-source simulation and programming environment.



Link @ Git

All code is based on **middleware ROS (first version) and developed with C++ and Python**. For the preparation of the race, the simulation environment, for the training or design of the locomotion framework, is provided on a **Git repository** based on **Gazebo (3D robotic simulator)**. In order to create and test features and behaviours before the competition inside the simulator, it will be possible to reproduce the behaviour of the installed sensors (e.g. Lidar, RGB camera, Depth camera) using C++ or Python. The developed code will be then uploaded on the robotic hardware provided by EagleProjects.

For the preparation of the race, the simulation environment, for the training or design of the locomotion framework, is provided on a **Git repository** based on **Gazebo (3D robotic simulator)**. In order to create and test features and behaviours before the competition inside the simulator, it will be possible to reproduce the behaviour of the installed sensors (e.g. Lidar, RGB camera, Depth camera) using C++ or Python. The developed code will be then uploaded on the robotic hardware provided by EagleProjects.

Rules of the game

The RDRC consists of two "sub-challenges." If a team participates in both challenges, their ultimate ranking will be determined by the total score they earn in both sub-challenges. In the case of the Locomotion track (challenge 2), the robot is meant to be tele-operated. However, if the robot successfully completes the challenge in a fully autonomous manner, a x2 score multiplication factor will be applied. The score for each obstacle is determined based on its objective difficulty. The time to accomplish the task will also be an evaluation factor. For the Navigation Track (challenge 2), the final score is calculated by summing the points earned for each object, with the objective difficulty in the object detection determining the number of points received. The time taken to complete the task will also be evaluated.

Challenge 1 (Locomotion Track)

The aim of this challenge is to test the robot's ability to navigate various obstacles from a point A to point B along a designated path in the shortest time possible.

- The arena is enclosed by walls, preventing the robot from exiting.
- Participants are provided with a 3D map of the arena when they enrol in the challenge.
- There is a predefined path to follow that involves overcoming obstacles of different levels of difficulty.
- Some obstacles like stepping stones, debris, and pieces of furniture, are strategically distributed to challenge the robot's ability to navigate.
- Other obstacles include stairs, ramps with different slopes, tunnels, slippery terrain, and soft terrain (such as foam). Some obstacles require specific abilities, such as crawling under a low bar or jumping over a high step.



- Penalties are given if an obstacle is skipped or if there are collisions with the robot's body. However, collisions with any part of the leg (such as the shin) do not result in penalties.
- If the robot touches the walls, there is a time penalty (10s subject to change)
- If the robot falls, it needs to restart from the beginning of the circuit.
- Each team will have 2 attempts (subject to change) of race and each attempt lasts at most 15 minutes (subject to change). Each attempt must start at the starting position. The team's official score will be selected from the best attempt. Each attempt will be recorded in time. The record time of each team's attempt is measured from the time to take the first step and to the time it finishes the attempt.
- Each attempt is considered "finished" by successfully passing the finish line. If the robot deviates from the course, or malfunctions, is not able to sustain safe and controlled walking, or the team operator declares ending the attempt, this attempt is considered "not finished".
- The winning team shall be the team with the shortest race time if the team finishes the whole course overpassing all the obstacles (except for the final one that gives a bonus if over-passed). If the team does not finish the whole race track, the final location where the robot reached and the ending time is recorded. The final location is measured from the start line to the farthest point on the robot.
- For grading teams who did not finish the whole course will be compared with the position where it reached. If two or more teams reached the same position, the team with shorter race time shall be given higher ranking.
- The judging committee reserves the right to stop any team's attempt if considered dangerous or not following the guidelines.

Challenge 2(Navigation Track):

The objective is to explore an unknown maze environment in order to find given objects, in the shortest time possible. The robots will operate within a designated area of a maximum of 100 square meters, characterized by the following conditions:

- The maze is enclosed by walls, preventing the robot from exiting. No map of the maze will be disclosed to the participants that will have to implement their own exploration strategy.
- The maze floor can be cluttered with moderate obstacles.
- Three (subject to change) objects will be hidden in the maze to be discovered.
- The nature of these objects will be communicated to participants one week prior to the competition.
- Throughout the competition, each robot is allotted a fixed time (to be defined) to locate all objects within the arena. The robot that successfully finds the most objects or locates all objects in the shortest time will be declared the winner.

Software/Code Availability

We provide a locomotion framework (Wolf) for the simulation of the challenge.

For the locomotion track, the teams are expected to develop their own framework.

The code is available at the following link (https://github.com/ERF2024/dog_challenge)

Q&A sessions

Every two weeks, the challenge organizers will gather for a question-and-answer session to further refine baseline codes (if needed) and assess queries from the participants.

Requests related to this specific challenge shall be addressed to

- **Prof. Michele Focchi** (michele.focchi@unitn.it)
- **Dr. Andrea Pupa** (andrea.pupa@unimore.it)



2 Robotics in Agriculture



The Fruit Harvesting Challenge (FHC) is a competition supported by the companies **Unitec**, **Universal Robots**, **QB Robotics** that will sponsor the challenge by providing two robot arms and two robotic grippers (anthropomorphic soft hands).

As robotics is rapidly expanding in the agricultural context, especially for task automation, and orchard management is one of the less automated scenarios, the motivation of this challenge is to highlight possible solutions for fruit harvesting. The goal is to detect apple fruits directly on the tree

and reach them with a robotic arm in order to perform the picking manoeuvre.

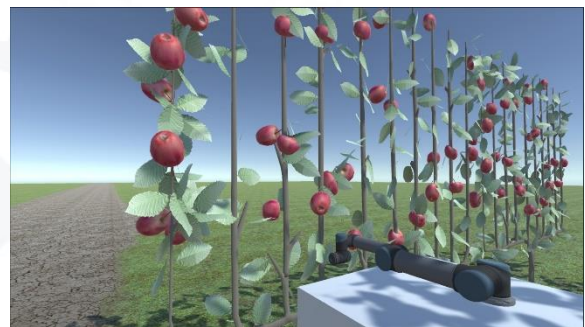
The challenge has great potential to lead the robotics community in technology advancement, nurture field engineers, and foster interactions with both farmers and technology companies, ultimately leading to the creation of practical services for the farming industry.



Description

The challenge requires to control a robotic arm to automatically collect apple fruits from a mock-up tree environment. Teams will be provided with a **Unity simulation environment and a dedicated ROS2 workspace** for training and testing the implemented solutions. The developed code will be then uploaded on the robotic hardware provided during the ERF.

The arm is equipped with a soft-robotic hand with a camera mounted to perform fruit detection and tracking. The camera will be an Intel **Realsense D435**. The teams are required to acquire images from the camera, detect at least one apple, reach it, grasp it and detach it from the mock-up tree.



Baseline Code. Baseline code is available for the fruit harvesting challenge through a ROS2 workspace repository. Instructions for ROS2 (Humble) installation can be found on the official website. Check the readme page on the repository for the provided packages description.

Simulation Environment. A simulation environment of the benchmark has been developed internally from the University of Bologna to use Unity simulation engine connected with Robot Operating System (ROS) version 2. This can be used to provide an accessible and reliable testing environment to facilitate both development and learning-based approaches.

The code will be available after challenge registration and will require Unity version 2022.3.9f1 installed on your development machine. Setup instructions will be provided together with the simulation package.



Link @ Git

Rules of the game

The challenge will target teams capabilities to implement the following process:

- Fruit recognition
- Fruit tracking and approaching
- Fruit picking



- Detach maneuver
- Harvested fruit placement on bin

The score will be primarily based on the number of picked fruits during a 5 minutes time frame.

The arena will be composed of a mock-up orchard with artificial trees and mock-up apples attached. The apples will have some predefined positions and placements that are supposed to be harder to reach or detect will provide increased score. The points acquired for each correctly picked fruit will be depicted directly on the fruit.



In front of the artificial orchard, two Universal Robot UR5 arms will be placed at a fixed position in order to create a small intersection of the two workspaces.

On the arm end-effector there will be a hand-shaped gripper, namely qb SoftHand Industry, to perform the actual fruit picking. The gripper is controlled by means of an open/close state.

Arms joints can be controlled both in velocity or position set-points.

The RealSense RGBD camera can be placed on a predefined fixed position at arm base or mounted near the gripper, within the last joint of the arms.

The teams are supposed to implement an apple detection (and tracking) system, control arm joint positions to reach the apples, perform picking maneuver and place picked apples inside a container placed between the two robotic arms.

The teams are supposed to implement an apple detection (and tracking) system, control arm joint positions to reach the apples, perform picking maneuver and place picked apples inside a container placed between the two robotic arms.

- The timer starts when the system is started for the first time, and after the deadline, each movement has to be stopped.
- If problems arise, it is allowed to manually restart the system. Timer will not reset, arms have to be restored to a safe “home” position with grippers opened before continuing. Any non-placed apples will be lost.
- Total score will be acquired by summing the points of each apple inside the container after the time limit.
- Penalty applies if the arms collide. In this case, the test may be halted and the session stopped.
- An expert jury may evaluate the different solutions to provide special bonus points.



Q&A sessions

Every two weeks, the challenge organizers will gather for a question-and-answer session to further refine baseline codes (if needed) and assess queries from the participants.

Requests shall be addressed to:

- **Dr. Dario Mengoli** (dario.mengoli2@unibo.it)
- **Dr. Andrea Pupa** (andrea.pupa@unimore.it)



3 Robotic Manipulation in Manufacturing

The Robotic Manipulation in Manufacturing (RMM) is a competition supported by the companies [ABB](#) and [Bonfiglioli](#) that will sponsor the challenge by providing a robot arm + components of a commercial gear-reducers to be partially assembled. The challenge is organized by the *Scientific Committee of the Doctoral School in Robotics and Intelligent Machines*, that previously engaged several PhD students'

Power and productivity
for a better world™ **ABB**

Bonfiglioli



teams in a task requiring the automated manipulation of a dice. Building up on this previous experience, the RMM has evolved to an “industry-driven” challenge resembling a real-world scenario in which multiple components of a real epicyclic gear train are assembled by means of a robotic arm. In order to allow the use of a low payload Cobot, the components to be manipulated will be a 3D printed plastic copy of a commercial mechanical system produced by *Bonfiglioli*. In practice, the RMM has been conceived to promote pivotal concepts in automated manipulation for Smart Manufacturing and Factories of the Future.

Description

The challenge requires the assembly of a 3D printed gearbox employing an industrial Cobot. You will manipulate the end effector of a serial robot (ABB Yumi single arm) equipped with a specially designed gripper. A simulation environment of the benchmark has been developed internally from the Milan Polytechnic to use Gazebo simulation engine connected with Robot Operating System (ROS), to facilitate both development, accessibility and learning-based approaches.



Link @ Git



Rules of the game

Objective: Design and implement a comprehensive system utilizing two Yumi robotic arms to efficiently assemble planetary gear systems. Scoring is contingent on the successful assembly of parts within a 5-minute timeframe.

Task Breakdown:

1 Gear Recognition:

- Develop a robust gear recognition system utilizing RealSense RGBD camera.
- Assign predefined scores based on the complexity of detecting and reaching each gear.

2 Gear Tracking and Approaching:

- Formulate a tracking algorithm to monitor the movement of recognized gears.
- Calculate optimal paths for the Yumi robot arms to approach each gear.

3 Gear Assembly:

- Control the gripper on the end-effector to execute precise gear assembly maneuvers.
- Ensure the gripper is in the correct open/close state as required during the assembly process.

4 Detach Maneuver:

- Implement a detachment maneuver to release the assembled gear securely, minimizing the risk of damage.

5 Assembled Gear Placement:

- Program the Yumi robot arms to accurately position the assembled gears in containers strategically placed between the two robotic arms.

Scoring:

- Award points directly on the gear based on the difficulty level of assembly.
- Sum the points acquired for each correctly assembled gear within the 5-minute time frame.



Timer and System Restart:

- Initiate the timer upon system activation.
- Allow manual system restart in case of issues, with the timer persisting.
- Mandate the restoration of arms to a safe "home" position with grippers opened before resuming.

Score Calculation:

- The total score is the cumulative sum of points for each gear correctly placed inside the designated container after the time limit.

Penalties:

- Impose penalties for collisions between the robot arms.
- Halt the test and conclude the session in the event of a high speed collision.

Expert Jury Evaluation:

- An expert jury will evaluate various solutions and may grant bonus points for:
 - Most innovative approach
 - Most reliable performance
 - Thinking out-of-the-box solutions

Note:

- The setup comprises two Yumi robotic arms, an RGBD camera, and containers containing individual gear components.
- The challenge aims to showcase teams' proficiency in designing a comprehensive system that balances speed and accuracy in gear assembly.
- Safety is paramount, and penalties are enforced for collisions.
- Bonus points are awarded to encourage teams to explore inventive and creative solutions.
- The final score reflects the cumulative points achieved for successfully assembling gears, factoring in their respective difficulty levels.

Q&A sessions

Every two weeks, the challenge organizers will gather for a question-and-answer session to further refine baseline codes (if needed) and assess queries from the participants.

Requests shall be addressed to:

- **Prof. Andrea Maria Zanchettin** (andreamaria.zanchettin@polimi.it)
- **Dr. Andrea Pupa** (andrea.pupa@unimore.it)



4. Robotic Drone Contest

The Robotic Drone Contest (RDC) is a competition supported by the company [Leonardo](#), as an open innovation project that can drive innovation and stimulate new ideas towards increasingly complex requirements.



Among these, multiple opportunities for collaborative heterogeneous robotic platform systems are envisaged, such as the inspection of areas affected by natural disasters or sites that are difficult for humans to access, and/or patrolling activities. The purpose of the project is to integrate a system of heterogeneous robotic platforms, both UAVs (Unmanned Aerial Vehicles) and eventually UGVs (Unmanned Ground Vehicles), capable of moving autonomously in unknown environments and without GNSS (Global Navigation Satellite System) signal and providing information about the environment in which they operate to a ground control station where the human operator is present.

Description

Each team participating the contest must be equipped with at least 1 UAV platform but could also expand its fleet with UGVs or other UAVs. The sensor data must be used jointly and interpreted on board the UAVs and UGVs. The information acquired by the various agents must be shared with the human operators at the ground control station in order to provide them with pre-processed data, such as the mapping of the environment and the presence of targets.

The use of ROS2 and therefore DDS will be mandatory.

In addition, all the software must use containerization with Docker. The system must be equipped with a human-machine interface (HMI). Through this HMI, operators can view the information related to the mission. For example: map of the environment, status of the robots (battery charge, flight mode), images of the targets identified by the onboard cameras, interact with the robots to issue high-level commands (e.g., start and end mission). Each team must use an interface that could allow the management of the mission by an operator external to the team, such as a judge.

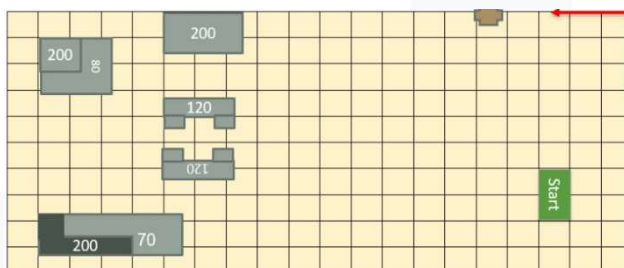


The environment is partially unknown & a partial map will be provided. The characteristics will be:

- Nets for fencing football and soccer fields
- UV-stabilized fencing nets
- Wire diameter 2 mm
- Square mesh 13x13 cm
- Lightweight water-repellent fencing nets
- Weight per square meter 30 grams
- Minimum net height 3 meters

The field will simulate an urban scenario consisting of non-magnetic obstacles up to 3 meters in height, with a minimum passage clearance of 1 meter for UAVs and 0.8 meters for UGVs. In addition to the obstacles indicated on the map, there will also be obstacles in unknown positions. These obstacles can range in height from 0.6 meters to 3 meters with a maximum diameter of 1 meter. In addition, there will be objects that will constitute targets to be identified by the robotic system. These targets, in an unknown quantity, will have minimum dimensions of 0.1 meters and maximum dimensions of 1 meter and prismatic and/or pyramidal shapes.

On each target will be one and only one Aruco [reference library: <https://tn1ck.github.io/aruco-print/>]. The Aruco can be placed on either the top or side faces of the target and will be 0.1x0.1 m in size. Each team will need to provide their own router to connect the agents with the ground station.





Rules of the game

At the start of the round, each team will receive a list containing the ID of the Aruco present on the targets to be identified within the competition field. Inside the arena, there will be more targets, but only those indicated in the list will contribute to the final score. UAVs and UGVs, autonomously and without GNSS signal, must patrol the competition field, reconstruct the map, identify and locate the fixed targets present in the arena. The end of the mission will be given to the system by the Ground Control Station. Here are the main rules:

- Teams will have 3 rounds of 20 minutes divided into 3 days.
- Within the round, it will be possible to carry out multiple tests and the score of the test with the highest score will be considered. At each start of a new mission, the map reconstruction performed in the previous test will be deleted.
- It will be possible to communicate with the UAV and UGV from the GCS only to allow the exchange of: *i)* Sending Start/end signal to the team, *ii)* Receiving video and images, *iii)* Receiving Log, *iv)* Receiving update of the map status

Evaluation Criteria

At the end of the competition, the team with the highest total score will be awarded.

Score calculation:

- Each target will be assigned a score. The score will be validated when a photo containing the information about the identified target (readable Aruco) and its coordinates (x; y) is sent to the GCS. The time must be recorded for each send.

The total score will be determined according to a formula that will be shared before the competition and, for the purpose of calculation, it will be mandatory for each team to deliver the mission logs (PX4) at the end of the round and a text file showing the times of each mission phase and the information necessary for Leonardo to calculate the final score. The total mission time will have a weight on the final score: the total score of the teams that will be able to complete the mission correctly (identification of all targets) in less than 3 minutes will be increased by 25%. The following assumptions apply:

- During each round, the team may perform multiple tests and takeoffs.
- At the time of the UAV takeoff, no one will be allowed to be inside the field for security reasons.
- To ensure the safety of the drone and the people present during the competition, it will be mandatory to provide a command and control system to be activated in case of need, such as to switch the UAV from autonomous mode to a piloted mode.
- The tasks will be different for each round and will vary from test to test, as well as the arrangement of unknown objects & the number of targets that will be communicated by the judge.
- During all rounds, the safety pilot will be the only one to be in Visual Line of Sight (VLOS) with respect to the drone.

Q&A sessions

Every two weeks, the challenge organizers will gather for a question-and-answer session to further refine baseline codes (if needed) and assess queries from the participants.

Requests shall be addressed to:

- **Dr. Nicola Mimmo** (nicola.mimmo2@unibo.it)

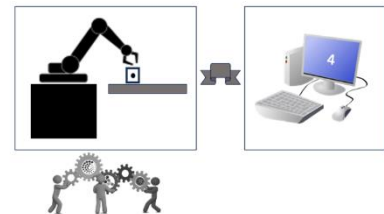


5 Robot navigation in an urban environment

The hackathon will be a competition between students from secondary schools in Italy. With educational and social purposes, the theme of the hackathon will be navigation in an urban environment.

Description

The ERF is one of the most influential events for the robotics and artificial intelligence community in Europe. For the first time in Italy, the ERF will take place in Rimini from 13 to 15 March 2024. It is the meeting point for engineers, academics, entrepreneurs, investors, as well as end-users and policymakers in the field of robotics from all over Europe and beyond. The ERF is promoted by euRobotics whose main mission is to strengthen the competitiveness and ensure the industrial leadership of manufacturers, suppliers and end-users of systems and services based on robotic technology.



The Robotic Kit

Each group of students will be equipped with a robotic kit, equipped with an Arduino programming board, dedicated to the creation of ground vehicles with sensory capabilities typical of the most advanced cars currently on the market. The kit contains

- 1 Camera
- 4 Motors
- 1 Position Servo
- 1 Line Sensor
- 1 Sonar
- 1 Arduino
- 1 Chassis

For details, see [Smart Robot Car Kit V4.0 \(With Camera\) – ELEGOO Official](#). For the development of the project, students will be able to take advantage of the support material available on the page [ELEGOO Established Sponsorship with Nuhlis to learn STEM knowledge and – ELEGOO Official](#).



The Hackaton

The robotic challenge will consist of tackling a path, in stages, in which all the on-board sensors (radar, camera, and line sensor) will be involved. The figure shows a facsimile of the test scenario.

Preparation and execution

Students will be able to develop their own robotics solution at their home institutions with the support of their teachers and the material available on the internet, books and manuals. The competition will be held in person on one of the days of the forum.

Rules of the game

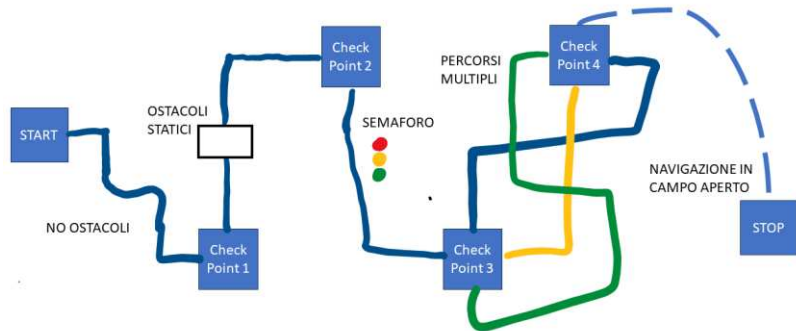
Students are not obliged to use the chassis supplied with the kit and any replacement must be made with recycled materials. There are no limitations on the mechanical design of the vehicle. For example, the choice of the number of drive and steering wheels is left free as well as the positioning of sensors and actuators. The combination of actuators and sensors is left free as long as it is drawn only from the kit supplied. The route has been divided into stages to allow the execution of individual independent tests. In this way, failure to



perform one stage will allow the execution of the following ones. For each stage, a score between 0 and 10 will be assigned, which will take into account the success and execution time of the trades.

The stages have been designed to pursue the following objectives:

1. Trail tracking. Rating index = travel time. Possible robotic solution = use the position servo to steer the front wheels. Use the line sensor.
2. Avoid fixed obstacles (local route replanning). Rating Index = Replanning Capability. Possible robotic solution = use sonar/camera to find an unobstructed path.
3. Obey traffic lights by not changing your route (local speed replanning). Rating Index = Replanning Capability. Possible robotic solution = use the camera to recognize the colors of the traffic light.
4. Choose from multiple paths. Rating index = choice of the right path (randomly assigned). Possible robotic solution = use the camera to identify the line without getting confused at intersections.
5. Navigate by sight in the absence of signs. Rating index = ability to get to the finish line. Possible robotic solution = rotate the camera to determine the position of the finish line.



Q&A sessions

Every two weeks, the challenge organizers will gather for a question-and-answer session to further refine baseline codes (if needed) and assess queries from the participants.

Requests shall be addressed to:

- **Dr. Nicola Mimmo** (nicola.mimmo2@unibo.it)