Confidentiality note:

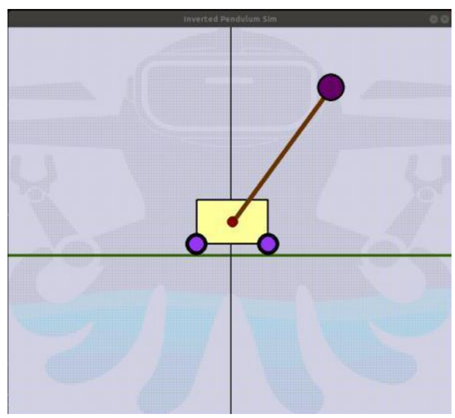
This document is meant for the internal evaluation of a job candidate for a position at Octobotics Tech Pvt. Ltd.

- It must not be shared with anybody other than the candidate for the job position mentioned on the cover page. The candidate must not make it available to the public or share it with anybody else.
- The applicant must discard this document once the intended purpose has been met.
- Please reference any third-party materials utilized in the submission, such as links to websites, movies, code, and so on.

Problem description

The assignment consists of an inverted pendulum mounted on a motorized cart. The inverted pendulum system is an example commonly found in control systems textbooks and research literature. Its popularity derives partly from the fact that it is unstable without control; that is, the pendulum will simply fall over if the cart isn't moved to balance it. Additionally, the dynamics of the system are nonlinear. The objective of the control system is to balance the inverted pendulum by applying a force to the cart that the pendulum is attached to. A real-world example that relates directly to this inverted pendulum system is the attitude control of a booster rocket at takeoff.

In this scenario, we'll look at a two-dimensional problem in which the pendulum can only move in the vertical plane, as indicated in the diagram. The force that pushes the cart horizontally is the control input for this system, and the outputs are the angular position of the pendulum and the horizontal position of the cart.



- The inverted_pendulum_sim package from the Octobotics_Coding_Assignment repository is heavily used in this problem.
- The package includes a motorized cart with a pendulum that can move in a 2D space and respond to control force commands.
- To learn how to communicate with an inverted_pendulum_sim, look through all of the APIs (ROS services and topics) given by the node. (README.md)
- You'll be given three subgoals, each with a different level of difficulty.
- Each goal has its own set of success criteria and is typically built upon the preceding one.
- To answer the problem, you may use any language you choose (C++, Python, etc.).

Goals:

Goal1: Creating a controller package

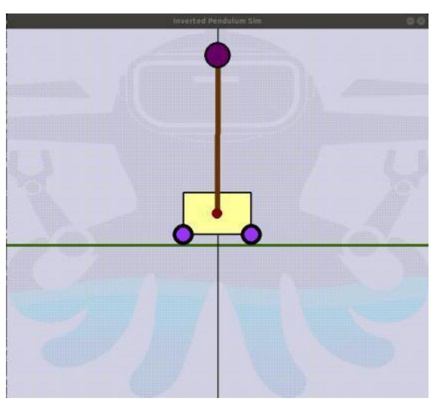
- Create a GitHub account and fork the Octobotics_Coding_Assignment repository.
- In the src directory, create a ROS package named inverted_pendulum_controller with the appropriate dependencies.
- Run the inverted pendulum sim using the initial parameters listed below.
  pendulum weight = 2kg, cart weight = 0.5kg, pendulum length = 300units, cart position = in the centre, pendulum orientation = vertical down pendulum length = 300units)

Goal 2: Send control input to the pendulum

- Provide a sinusoidal force input to the cart so the system oscillates in the visible window.
- Record a video with changing frequencies and amplitudes of oscillations.
- Include graphs of cart velocity, acceleration, as well as pendulum angular velocity, and acceleration with your videos.

Goal3: Balance the inverted pendulum

- Create and implement a control algorithm, such as PID/LQR, to balance the pendulum in an inverted position.
- You can learn about system modeling and controller implementation by visiting this page.
- Fine-tune the controller parameters for optimal performance, and describe the process.
- Document the results with graphs and videos.



Submission Instructions:

Code:

- Create a free GitHub account and fork the Octobotics Coding Assignment repository into your account.
- Please send the repository to ishan.b@octobotics.tech, cc: charith.r@octobotics.tech.
- Provide relevant instructions for setting up the environment and executing the project.

Videos: For each goal, create one or more videos (screen recordings).

Report:

- Summarize the problem-solving strategy.
- Explain any assumptions you made.
- Explain why you chose the algorithms/tools you did to solve the challenge.
- Display the results with graphs or videos.
- Explain what could have been done better if the submission deadline had not been imposed.
- Send the report in PDF format through email, along with any videos that may be attached.

Additional notes:

- If you do not have experience with tools/algorithms, please include this in your report. The evaluation will be carried out as planned. In such instances, view it as a challenge and do your best.
- The report is equally weighted in your evaluation, so make sure you spend enough time on it. If you have any questions or concerns about the assessment, drop an email to charith.r@octobotics.tech

Metrics of evaluation:

Overall knowledge of the problem statement, Justification for approach selection, choice of tools/protocols, Correctness of solution, Coding skills (usage of OOP, modularity, etc.), Coding style (PEP 8, etc.), Code documentation.