

Vast Take-Home Coding Challenge

Objective:

You are tasked with developing a simulation for a lunar Helium-3 space mining operation. This simulation will manage and track the efficiency of mining trucks and unload stations over a continuous 72-hour operation.

Key Components:

- **Mining Trucks:** These vehicles perform the actual mining tasks.
- **Mining Sites:** Locations on the moon where the trucks extract Helium-3. Assume an infinite number of sites, ensuring trucks always have access to mine without waiting.
- **Mining Unload Stations:** Designated stations where trucks unload the mined Helium-3. Each station can handle one truck at a time.

Operation Details:

- There are (n) mining trucks and (m) mining unload stations.
- Mining trucks can spend a random duration between 1 to 5 hours mining at the sites.
- It takes a mining truck 30 minutes to travel between a mining site and an unload station.
 - Assume all trucks are empty at a mining site when the simulation starts.
- Unloading the mined Helium-3 at a station takes 5 minutes.
- Trucks are assigned to the first available unload station. If all stations are occupied, trucks queue at the station with the shortest wait time and remain in their chosen queue.

Simulation Requirements:

- The simulation must be configurable to accommodate various numbers of mining trucks (n) and unload stations (m).
- Calculate and report statistics for the performance and efficiency of each mining truck and unload station.
- The simulation represents 72 hours of non-stop mining and must execute faster than real-time to provide timely analysis.

Language and programming paradigms:

Please implement this project in C++. Please leverage OOP where it is appropriate.

Goal of the exercise:

The primary goal of this challenge is to demonstrate your professionalism as a software engineer. This process is designed to mimic a real-world scenario, including design, implementation, and design review. You will be evaluated based on various skills, including:

1. **Communication:** Clear and concise explanations of your code and design.
2. **Documentation:** Providing well-documented code and explanations.
3. **Code Cleanliness:** Writing clean and organized code.
4. **Code Deployment:** Demonstrating your ability to deploy and manage code.
5. **Testing:** Implementing appropriate testing strategies.

Things to avoid:

It is not the objective to spend an excessive amount of time on this challenge or create a fully developed system. Feel free to include pseudocode (in comments) to explain what you would do if given more time or resources. The focus is on showcasing your problem-solving and coding skills within a reasonable time frame.

How to Submit:

Submit your code and any accompanying content, such as data or results, using GitHub or Bitbucket. Please email us a link to your submitted code when you are ready for us to review it.

Questions:

Please feel free to ask any clarifying questions about this assignment via email with your recruiter.