**CSE3063 LAB**

**Name:** Preyash  
**Registration Number:** 20BPS1022

**Black Box Testing**

**Use case: Rover Navigation Control**

1. **Steering Commands:**
   1. Test Input: Send various steering commands (e.g., left, right, forward, backward).
   2. Expected Output: Rover moves in the specified direction accurately.
2. **Waypoint Setting:**
   1. Test Input: Set waypoints at different coordinates on the planetary surface.
   2. Expected Output: Rover navigates to the specified waypoints correctly.
3. **Monitoring Movement:**
   1. Test Input: Monitor rover's movement in real-time.
   2. Expected Output: Receive accurate and continuous updates about rover’s position and status.
4. **Invalid Commands:**
   1. Test Input: Provide invalid or unsupported commands (e.g., nonsensical directions).
   2. Expected Output: System handles invalid inputs gracefully, perhaps by providing an error message.
5. **Load Testing:**
   1. Test Input: Simulate a large number of navigation commands in a short time.
   2. Expected Output: The system should handle the load and process commands within an acceptable timeframe.
6. **Interrupt Testing:**
   1. Test Input: Interrupt navigation with emergency stop commands.
   2. Expected Output: Rover should immediately stop all movement and respond to emergency commands effectively.
7. **Power Loss Simulation:**
   1. Test Input: Simulate a power loss during navigation.
   2. Expected Output: The system should have mechanisms in place to resume operation or enter a safe state when power is restored.
8. **Localization Accuracy:**
   1. Test Input: Test the rover's ability to accurately determine its position.
   2. Expected Output: The rover should accurately determine its location using sensors and GPS data.
9. **Network Disconnection:**
   1. Test Input: Disconnect the communication link between the control center and the rover.
   2. Expected Output: The system should handle communication loss gracefully, perhaps by attempting reconnection or entering a safe mode.
10. **Sensor Failures:**
    1. Test Input: Simulate failures in sensors (e.g., camera, distance sensors).
    2. Expected Output: The system should handle sensor failures and continue navigation using available sensor data or enter a safe mode.

**Use case: Rover Obstacle Detection and Avoidance**

**Obstacle Detection:**

- Test Input: Place obstacles of various sizes and shapes in the rover's path.

- Expected Output: The Rover's sensors should detect obstacles and the rover should navigate around them without colliding.

**Hazard Detection:**

- Test Input: Simulate hazardous terrain (e.g., steep slopes, loose soil).

- Expected Output: Rover should identify hazardous conditions and adjust its path to avoid these areas.

**Dynamic Obstacle Avoidance:**

- Test Input: Introduce moving obstacles in the rover's path.

- Expected Output: Rover should dynamically adjust its path in real-time to avoid moving obstacles.

**Testing Sensor Range:**

- Test Input: Place obstacles at the maximum and minimum detection range of the rover's sensors.

- Expected Output: Rover should detect obstacles within its specified sensor range and navigate around them accordingly.

**Edge Cases Testing:**

- Test Input: Test scenarios where obstacles are placed near cliffs or other dangerous terrains.

- Expected Output: Rover should recognize the danger and avoid approaching edges to prevent falling.

**Simulated Weather Conditions:**

- Test Input: Simulate adverse weather conditions (e.g., dust storms, low visibility).

- Expected Output: Rover's sensors and algorithms should adapt to the reduced visibility and still avoid obstacles effectively.

**Sensor Redundancy Testing:**

- Test Input: Simulate failure in one of the obstacle detection sensors.

- Expected Output: Rover should compensate for the sensor failure by relying on other sensors and still avoid obstacles successfully.

**Testing Response Time:**

- Test Input: Introduce sudden obstacles in the rover's path.

- Expected Output: Rover should detect the obstacle quickly and change its path promptly to avoid collision.

**Use case: Rover Sample Collection and Analysis**  
  
**Targeted Sample Collection:**

- Test Input: Specify various target locations for sample collection.

- Expected Output: Rover should accurately navigate to the specified target locations and collect samples from those locations.

**Random Sample Collection:**

- Test Input: Instruct the rover to collect samples from random locations.

- Expected Output: Rover should autonomously select random locations, collect samples, and store them for analysis.

**Sample Identification:**

- Test Input: Collect samples of different materials (e.g., rock, soil) and unknown substances.

- Expected Output: Rover's analysis tools should identify and categorize the collected samples correctly.

**Sample Preservation:**

- Test Input: Collect organic or delicate samples that require special preservation methods.

- Expected Output: Rover should handle and preserve delicate samples properly, ensuring their integrity during transport and analysis.

**Sample Volume Measurement:**

- Test Input: Collect samples of varying sizes and volumes.

- Expected Output: Rover's tools should measure the volume of collected samples accurately.

**Real-time Analysis:**

- Test Input: Perform real-time analysis of collected samples on-site.

- Expected Output: Rover's analysis tools should provide immediate data and insights about the composition of the collected samples.

**Sample Contamination Prevention:**

- Test Input: Implement protocols to prevent sample contamination during collection and analysis.

- Expected Output: Rover should follow contamination prevention procedures and ensure that collected samples remain pure and uncontaminated.

**Sample Retrieval Confirmation:**

- Test Input: Request the rover to confirm successful sample retrieval.

- Expected Output: Rover should provide confirmation signals or messages when samples are successfully collected and stored.

**Multiple Sample Collection:**

- Test Input: Instruct the rover to collect samples from multiple locations in a sequence.

- Expected Output: Rover should navigate to each specified location, collect samples, and store them without confusion or mix-up.