**Team Name : ShowStoppers  
Project Title: Project Proposal for a Treasure Hunt Game**

**1. Interactive 2D Treasure Hunt Game with AI Algorithms**

**2. Objective:**

The goal of this project is to create an engaging 2D treasure hunt game where players or AI agents navigate a map to find hidden treasures. The game will incorporate advanced algorithms for navigation, pathfinding, and decision-making to enhance the experience.

**3. Key Features:**

1. Single-Player Mode:

The player competes against time to locate the treasure on a predefined map. Obstacles and hints are placed strategically to make the game challenging.

2. Two-Player Mode:

Players compete against each other to reach the treasure first. Includes adversarial AI for single-player mode where the player competes against an AI-controlled agent.

3. Dynamic Map Generation:

Randomly generates maps for every new game to ensure variety.

4. Pathfinding Algorithms:

A\* Algorithm for optimal pathfinding. BFS/DFS for alternative search strategies.

5. Interactive Hints System:

Players can discover hints like directions or distances to the treasure, adding strategic elements.

---

**4. Algorithms Used:**

1. Pathfinding Algorithms:

A Algorithm:\* For efficient navigation to find the shortest path to the treasure. BFS/DFS: For exploration and obstacle traversal.

2. Adversarial Search: Minimax Algorithm with Alpha-Beta Pruning: Used in two-player mode for AI decision making.

3. Random Walk: For randomized treasure placement on the map.

4. Dynamic Scoring: Implements a scoring system to evaluate players' performance based on efficiency and time.

**5. Technology Stack:**

Programming Language: Python

Development Tools:

Pygame: For 2D game development.

PyCharm: For coding and debugging.

Algorithms: Pathfinding (A\*, BFS, DFS), Adversarial Search (Minimax).

Graphics and Assets: Basic sprites for characters, treasure, and obstacles.

---

**6. Project Phases and Timeline:**

1. Phase 1: Planning and Design (Week 1) Define game rules, map structure, and algorithm requirements. Create sketches or mockups for the game layout.

2. Phase 2: Development (Weeks 2-3) Implement the game map and treasure placement. Integrate A\* Algorithm and BFS/DFS for player navigation. Add obstacles and hints system.

3. Phase 3: Adversarial AI Development (Week 4) Implement adversarial AI using Minimax with Alpha-Beta Pruning. Test and refine the two-player mode.

4. Phase 4: Testing and Optimization (Week 5) Conduct gameplay testing to ensure algorithms perform correctly. Optimize game performance and fix bugs.

5. Phase 5: Deployment (Week 6) Finalize the game with UI enhancements. Prepare documentation and user guide.

---

**7. Expected Outcome:**

The project will result in a fully functional 2D treasure hunt game that demonstrates the practical application of search and decision-making algorithms. It will be both entertaining and educational, showcasing algorithmic efficiency in game design.

---

**8. Potential Enhancements (Future Scope):**

Add multiplayer support for online gaming.

Introduce dynamic difficulty adjustment based on player performance.

Use machine learning for AI behavior improvement.

---