Programming Assignment

Game Brief

Crimson Tactics is a 3D Tactics Game. The main gameplay is in the battlefield in isometric maps. Please refer to this image to get an idea of how the isometric battle view will look:

This is from Tactics Ogre: Let us Cling Together.

Assignment 1 – Grid Block Generation

Generate a 10x10 grid of Unity Cubes. Each cube should be a Gameobject and attached to it should be a script which has information about that particular tile.

After that, do a ray cast from the mouse. When the mouse hovers over the tile, you should read that tile's information using the ray cast and display the unit's position on the grid on a UI element.

Assignment 2 – Obstacles

Create a Unity Tool to generate obstacles on the grid generated in Assignment 1. The Unity Tool should have 10x10 toggleable buttons representing the grid. If the button is toggled on then that means that particular tile is blocked using an obstacle.

The editor tool should edit a Scriptable Object in the project. Basically, the obstacle data is stored in that scriptable object.

An 'Obstacle Manager' script should read the Scriptable Object and generate red spheres representing obstacles on top of the grid created in Assignment 1.

Assignment 3 – Pathfinding

Generate a player unit on the map. The player should be able to move to any selected tile on the grid generated in Assignment 1 and should not be able to move to grids with Obstacles if Assignment 2 was attempted. Please don't use Unity Pathfinding for this. A grid-based pathfinding algorithm is expected. Please show movement when moving from one tile to another. Input should be disabled while the unit is moving.

Assignment 4 – Enemy AI

Generate an enemy unit. The enemy unit's objective is to move closer to the player unit. It should move using the same algorithm used in Assignment 3 or with Unity Pathfinding if Assignment 3 was not attempted. The enemy should attempt to move to one of the 4 adjacent tiles next to the player's tile. Once it reaches the desired tile, the unit should stay still until the player unit moves. This should be done following proper OOP concepts. The 'Enemy AI' script is expected to inherit from an 'AI' interface.

Miscellaneous

1. The Applicant is advised to present the project in an aesthetically pleasing manner. They may use any assets they see fit for this.
2. The Applicant is required to write comments throughout the code.
3. The Applicant is expected to not use Generative AI like ChatGPT to generate the scripts for this assignment. **Note**: This is fairly easy to check for.
4. After finishing the assignment, the applicant is expected to upload everything to a public GitHub repository.