/\*NOTES\*/

You MUST have Blender installed on your system.

The main components for this short demo are contained inside the folder “TileGameplay”

**Setup:**

Determine how big you want the board to be. The example scenes are 20 x 12 (240 tiles). If you expand the TileBoard game object you’ll see each tile and how they are setup/numbered. They must be numbered properly – one misnamed tile will disrupt the movement algorithms of each character. Make sure you know how big you want the board to be – if you decide you want to make the board bigger later, you will have to go back and rename every tile.

Once the board is setup and tiles properly named, set the TileBoardWidth and TileBoardLength variables of the BoardManager script on the LevelManager game object. If done properly, BoardManager will find each tile and place them into an array for later reference (important for character movement and board statistics).

**Adding Characters:**

You can have as many characters as you want. If they are a playable character, give them the “Player” tag. If they are an enemy, give them the “Enemy” tag. Tiles cannot have more than one character/object at a time. Just place the character off screen, and set their StartPosition variable – this refers to a tile number. If the startPosition is 10, they will be put on Tile10 at startup.

**Tiles/board:**

Tiles cannot have multiple objects. If a tile is occupied, characters cannot move there. Characters cannot advance past occupied tiles. You can any kind of board setup (not just a large block), but it must have boundaries (in the example scenes the black tiles are the boundaries – THESE MUST BE SET TO OCCUPIED in order to prevent characters from interacting with them). Boundaries do not have to be black, they just need to be set to ‘occupied’. You can turn off the rendering of each tile if you wish (though I would suggest doing that last – it is easier to select tiles during testing/debugging if you can see the tile). If you decide to only have the tiles render when showing a character’s valid moves, you will need to add the additional code to disable/enable the MeshRenderer component.

**Movement Algorithms:**

You can have any type of movement style for each character. The basic example on CharacterType1 uses the tile array to determine which tiles the player can move to (which is why tileBoardLength and tileBoardWidth are very important). When determining the movement style (which tiles are valid), think of the board as a coordinate system. For example, if I’m at Tile35 and I want to make an up movement in 1 unit valid, I would look at Tile55 which is equivalent to my current tile’s index (35) + tileBoardLength (20) = 55 = Tile55. Two units up would use tileBoardLength\*2 = Tile75.

**Static objects:**

You can place static objects on the board for aesthetic purposes, just be sure to set the tile(s) that contain that object to occupied (if you don’t want the player to be able to move to that tile).

**Scripts**

There are 4 main scripts:

**BoardManager**

BoardManager takes the tiles (searches by using their name) and puts them into an array for later reference. Tiles are named “Tile” + their number.

**CameraControl**

Controls the camera. This is where the player will select characters and tiles. This is also where the tiles are highlighted.

**CharacterType1**

This is a basic character. You can have multiple character types. This is where the character’s data should be kept (e.g. health, movement styles, etc.). CameraControl uses SendMessage to communicate with character scripts in order to make them move, and to show/unshow their valid movement tiles.

**GameTile**

This is on every GameTile prefab. This contains important data about the tile. Is the tile occupied? Is the tile occupied by a player or enemy? Can a character move here? This also contains the different materials (colors) associated with each tile – these change depending on what the player is doing (e.g. if the tile is a valid movement tile, the color of the tile will be blue).

**Scene notes:**

**TestScene1:**

Basic setup with 3 playable characters.

**TestScene2:**

Basic setup with an altered board. tileBoardLength and tileBoardWidth are still the same as in TestScene1 (20x12), but the boundaries have been changed, and the renderer component of certain tiles have been turned off.

**TestScene3:**

Same board but with static building objects and a terrain. The tiles on the edge of the buildings are set to occupied to prevent the character from interactig with tiles beyond those edge tiles.