Navigation Mesh

Lab:

- Explain the concept of a navigation mesh and how it can be used in a game environment for character movement.
- Develop a waypoint system on a navigation mesh and program an AI to move from one goal location to another.
    - Use the tank environment from previous lab.
    - Add extra obstacles
- Demonstrate the setup of a navigation mesh agent and describe the elements that provide differentiation between agents.
- Develop an AI that will follow the player around a game environment that has had a navigation mesh applied."

Nav Mesh intro

Originally known as Meadow Maps, this method was developed for robotic applications in the mid-1980s. By the 2000s, it gained popularity in video game development, allowing AI agents to navigate complex environments efficiently. A NavMesh breaks down a 3D model into a series of polygons, assigning a cost to each. This cost helps the pathfinding algorithm determine the most efficient route for an AI agent.

Nav Mesh functionality in Unity

Agents

- Defines AI agents and their properties.
- Each agent has four key parameters:
    - **Radius** (determines the width of the character).
    - **Height** (assumed when navigating flat surfaces).
    - **Slope Limit** (maximum gradient the agent can traverse).
    - **Step Height** (maximum stair height the agent can climb).
- These values determine where a character can and cannot go.

Areas Tab

- Assigns cost values to different parts of the NavMesh.
- The A* algorithm uses these values to determine the most optimal path.
- Unity includes three default area types:
    - **Walkable** (allows movement).
    - **Not Walkable** (blocks movement).
    - **Jumpable** (enables jumps).
- Developers can define custom areas, such as restricted zones or inaccessible regions.

Bake Tab

- Converts meshes into a NavMesh by dividing them into polygons.
- Uses the agent's properties (size, step height, slope limit) to determine where movement is possible.
- The default humanoid agent type is preconfigured, but Unity provides beta components for defining other agent types.
- Additional settings include:
    - Off-Mesh Links (enable jumping or dropping between gaps).
    - Voxel Size (controls mesh detail; larger voxels decrease accuracy).
    - Minimum Region Area (removes small, irrelevant nav areas).
    - Height Mesh (preserves stair-like structures rather than turning them into slopes).

Objects Tab

- Assigns specific area types to different scene elements.
- Controls which objects contribute to the NavMesh generation.
- Enables off-mesh links, allowing transitions between separate NavMeshes.