The **Underground Laboratory Generator** creates a level from details. Underground laboratories, caves, tunnels, metro, military base, underwater base... It works with any workpieces. determine the exits from the room and set the occupied area. Room in a random order will be to adjust to each other.

The generator works as follows:
A random room is selected from the list of prefabs and created in the center of the scene. Subsequent rooms are delivered to the scene and connected to a randomly selected unoccupied doorway of already created rooms. Verified by overlaying on each other. After installing the required number of rooms, all temporary platforms of rooms are replaced with randomly selected objects from the list of prefabs of doorways.

1. First room is created;
2. Substitute rooms relative to those already created, one by one;
3. Temporary doorways are replaced with prefabs of doorways.

How to use scripts with their own developments:

1. Create an empty object. Add the **Laboratory Generator** script;

2. Create prefabs for rooms (rooms, corridors and stairs). See below for how to set them up correctly;

3. Create prefabs for doorways;

4. Insert links to rooms and doors in the corresponding arrays in the **Laboratory Generator** script;

5. Check the **Generate On Start** box to generate a level when starting a scene in the game. If the generation is performed by a condition, you can call the level generation coroutine **Start Generation** of **Laboratory Generator** script.

Room-prefabs are created in the following order:

1- Create a mesh;

2- Add the **Cell** script. **Box Collider** will be added automatically (later it will be used as a trigger);

3- Adjust the **Box Collider** to the size of the room (this is necessary so that the rooms do not intersect during level generation). This is optional, because it happens automatically;

4- Add a **Collider** (for example, **Mesh Collider**), this is necessary to interact with characters and objects;

5- Set a platform-object for each exit from the room. The platform always looks outwards with the **Z-axis** (blue). In the future, the platforms will be a link between rooms and corridors, as well as a place for installing doorways;

6- Add platform-objects to the **Exits** list in the **Cell** script;

7- Set the layer (**Cells** in my case) only on the room-prefab (do not include child objects).

Setting up doorways:

1. The Doorway replaces the platform-object in the room-prefab, so the opening must be the same size or slightly larger than the opening in the room-prefab.

2. The **pivot** of the doorway and the placeholder must match.

3. Don't forget to set one of the collider-behaviors (for example, **Mesh Collider**) for interaction with character.

Recommendations:

1. the Prefab of the doorway should be made larger than the object-area (platform) in the prefab room, so that there is no gap between the doorway and the room. This may occur if the platform-object is slightly shifted when editing the room-prefab;

2. Preferably all of the room to mark as **Static**;

3. Global illumination set to **Source: Color**, grey color, directional light sources **Direct Light** to clean.

4. An object with the **Simple random** script marked **Enable children destroy** deletes child objects until the required number of child objects remains. To display correctly in View port, you can disable all child objects except one, so that the textures do not overlap if the objects overlap.