Applied Activity 6 – Minotaur's Labyrinth (Part 1)

This assignment consists of several C# files that together make a functioning (although pretty featureless) game. Use these files to create a Visual Studio solution as the starting point for your submission.

Refer to the grading rubric on Moodle for the marking breakdown.
This applied activity must be completed independently.
Do not use AI to code your answers (refer to my AI statement on Moodle).

Task 1 (100%)

You are given code that generates a 2D grid-based labyrinth where each grid represents a room. Most rooms are empty at this point except for the labyrinth entrance and a room that contains a magic sword. The purpose of this game is to navigate the labyrinth and find the sword and then escape the labyrinth.

However, an unnatural darkness pervades the labyrinth that inhibits the ability of light to permeate. The player must navigate the labyrinth in the dark relying on their senses of smell and hearing to determine what room they are in and what dangers lurk in nearby rooms.

How to Complete This Assessment:

1. Look over the code and run the game to get an idea of what the game's flow looks like. We will discuss the code together as it is reasonably extensive.
2. Look for comments labeled "**TODO: (A6)**". These are recommendations of where you should add code to implement the **Random start locations**. Feel free to add additional code in other places, if needed.
3. In the current version of the game, the player start location and sword location are hard-coded in the **LabyrinthCreator.cs** file. Your job is to randomize these two starting conditions. It is **highly recommended** we do our best to adhere to the **single responsibility principle** and make a **new class** to handle this task.
4. Keep in mind, the entrance should always be located on the edge of the 2D grid, and the sword location should not be adjacent or overlapping the start location.

How to Submit Your Applied Activity

Submit a zip file of all of your code. Be sure to Include some comments in your code that describe how you ensured your randomization process functions correctly (no bugs).