Odium's Mansion

Description

In the land of the Aldracs there is a mountain fortress called, *Odium's Mansion*, owned and defended by the evilest Aldrac that has ever lived. Aldracs have the excellent reputation of being the kindest and most hospitable of creatures, however, the Aldrac who owns the mountain fortress is so evil that his name must never be uttered, even in the softest of whispers, so it has been stricken from this document. We will only refer to him as, "the evil Aldrac", cursed be his name!

It is rumored that in this mountain fortress exists a chest of gold. Aldrac gold is of the purest to be had. It is said that the gold is as pure as an Aldracian's heart. Many-a thief has attempted to liberate the evil Aldrac, cursed be his name, of his treasure, but they had never been heard from again. You see, the Aldrac fortress has a dungeon, and in this dungeon is where the gold is hidden. The dungeon has 64 chambers and in some of these chambers there are traps in the form of bottomless pits. The only chambers that are known to be safe is the chamber in which you enter the dungeon and those directly adjacent to it.

The evil Aldrac himself stays in one of these chambers. It is rumored that he has gotten so fat from killing and eating greedy thieves and an occasional staff member, he cannot leave his chamber. Anyone whose has entered the evil Aldracian's chamber, cursed be his name, has never left it. Anyone who has fallen into a pit has never stopped falling.

There are "tells" (sensations) that give away the chambers with the pits. When in a directly adjacent chamber, you can feel a soft breeze at your feet.

The chambers directly adjacent to the evil Aldracian's chamber, cursed be his name, offer a hint to his presence as well. It has been said by staff who survived their employment, there is a stench in the chambers directly adjacent to the evil Aldracian's chamber, cursed be his name.

There is one more signal, perhaps the most important. It is very dark in the dungeon you see, but when you are in the chamber with the gold, you can see the dimmest glimmering light. The chamber with the gold has no pit and the gold is not in the chamber with the evil Aldrac, cursed be his name.

Task

You are a world-class thief as well, but you are smarter than the thieves who have disappeared. Armed with the awesome ability to program in C++, you will write a program that is a simulator for the Aldrac mountain dungeon. Every time the simulator is run it will randomly place pits throughout the dungeon except for the chamber in which you enter and those chambers directly adjacent.

The goal of the simulator is to train you to find the gold while avoiding the chamber with the evil Aldrac, cursed be his name, and those with pits. You have obtained some more valuable information to help build a better simulator. It was bought from an Aldrac-mountain servant who was missing an arm. She told you that there are exactly **12 pits** in the dungeon, but she does not know where. She also told you how to get into the dungeon. There is a ladder that descends to the first chamber; mentioned above. This is where your program will place you at the start of each simulation.

Activity

Implement this simulator. Follow all of the behaviors outlined here. Do not improvise on the output. Your output should look like the examples that follow.

The Simulator

Your simulator will follow this design exactly:

- Use at least one 2D array to simulate the dungeon. The dungeon has 64 chambers arranged in a 8x8 grid.

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

  Note: in the rest of this document, we will refer to the coordinates of a cell using its row and column. For example, (2,1) will refer to row 2, and column 1.
- Randomly distribute the 3 pits in the dungeon.
- Put the evil Aldrac, cursed be his name, in a random location in the grid.
  - The pits and the evil Aldrac, cursed be his name, must be in their own chambers.
- Your user, shown in the grid as a 'U', will start in the (0,0) location of the grid.
- The locations: (0,0), (0,1), and (1,0) will never contain a pit or the evil Aldrac, cursed be his name.
- You will use the following letters to signal any sensations that are experienced once you enter a chamber.
  S = Stench
  B = Breeze
  G = Glimmer
- You will accept as input the chars, **R**, **L**, **U**, and **D** to move the user right, left, up, down respectively.
- You will also accept the character, **P**, to pick up the gold when you are in the chamber with a glimmer.
- You will accept case-insensitive input. These input can be upper or lower case.
- You cannot exceed the boundaries of the grid. In other words, if the user is in chamber (0,0) it can only move to the right or down.
- Chambers that are directly adjacent means that they share a common wall. Being in a chamber diagonal to a one with a pit or the evil Aldrac, cursed be his name, will **not** reveal a breeze or a stench.
- When the dungeon's grid is displayed, there will be no indication of a breeze or a stench until the user enters the chamber in which it is sensed.
- Once a breeze or stench has been sensed, it will show in the chamber where it was sensed until the end of the simulation.
- When the grid is displayed, each chamber must have room for 4 characters. The first three are the possible sensations in the chamber, S, B, and/or G. The last of character will be a U if the user is in that chamber or blank otherwise.
- If the user moves into a room that contains a pit or the evil Aldrac, cursed be his name, the user dies and the simulation ends.
- If you pick up the gold, you must move to (0,0) to escape. Once there the simulation ends.
- With every move, the console should be cleared, and the updated grid displayed. I will provide the function for this. If you are building your program on a system other than Windows, Mac, or linux, you will have to figure out how to clear the console for that environment.

Design Notes

There are many ways to design this program. You could have a single 2D array of chars or ints. You can store a value in the element which represents the evil Aldracian's chamber, cursed be his name, and store another value for the pit and the user's position. You should store in each element of this array, a 0 if it is empty or one of three constant values to represent the user, the evil Aldrac, cursed be his name, and the pit.

You might want to create a parallel 2D array to keep track of where the user has been. This would give you the ability to display the sensations felt for each of the visited chambers. Once visited, each chamber should always display the sensations experienced until the simulation ends. You can calculate if there should be a stench, breeze, or glimmer in each element as you print out the board or you can keep parallel arrays to keep track of this. As always, be sure to be conservative with memory and processor use. Always try to code for efficiency.

I will provide two files called, *odiums_mansion.cpp* and *odiums_mansion.h*. In them I will include a function to clear the screen. Read the comments carefully to understand its use. I will also include the clearInstream function to be used to clear the std::cin stream.

Requirements

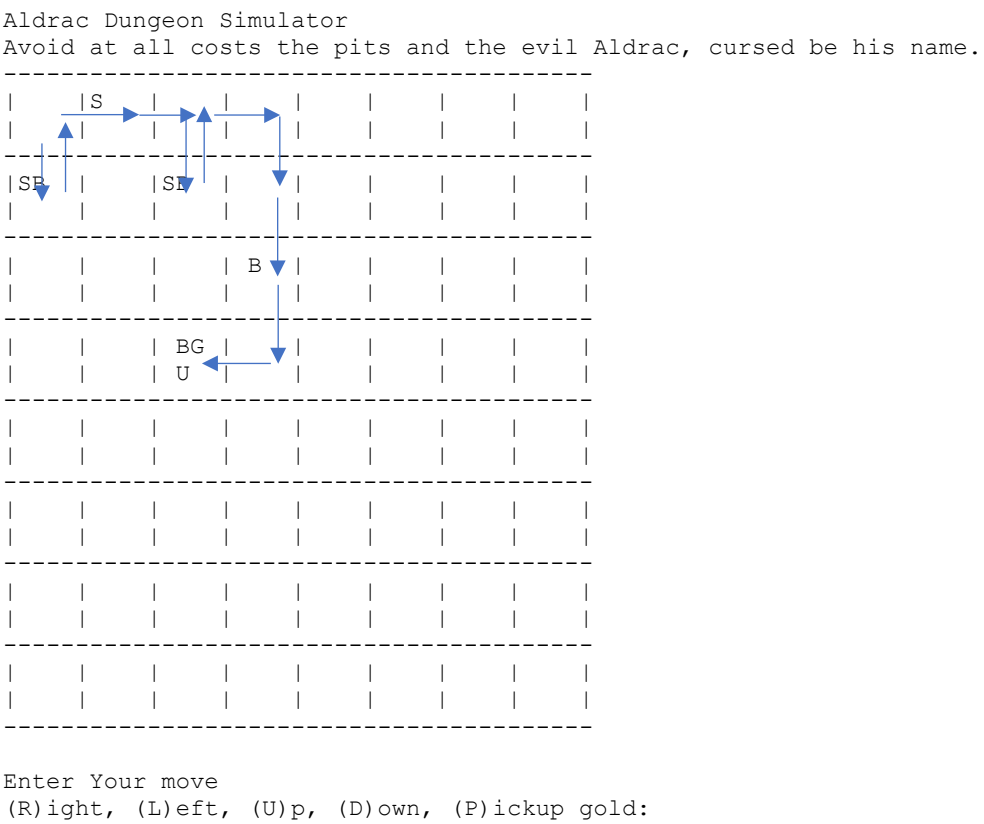
- The simulator must function as described in the "The Simulator" section.
- Your output must display the state of the simulator like this:

```
Aldrac Dungeon Simulator
Avoid at all costs the pits and the evil Aldrac, cursed be his name.
-----------------------------------------
|       |       |       |       |       |       |       |       |
|   U   |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------
|       |       |       |       |       |       |       |       |
|       |       |       |       |       |       |       |       |
-----------------------------------------

Enter Your move
(R)ight, (L)eft, (U)p, (D)own, (P)ickup gold:
```

  - I will make available a video of a simulation.
  - As you move through the dungeon more information will become available, specifically the sensations (tells) that you experience when you enter each chamber.
  - Note, the 'U' shows where the user is. The above example shows the start of the search.

- Here is an example after the users has moved:



  - The moves made in this example were: D, U, R, R, D, U, R, D, D, D, L
  - I've drawn arrows to show the course that was taken. **The arrows are not part of the program**.
  - The user is now at (3,2). All it needs to do is pick up the gold and navigate back the way it came.
  - Note: It is easy to see where the evil Aldrac, cursed be his name, is; at (1,1). Since (1,1) contains the evil Aldrac, cursed be his name, and a breeze was felt at (1,0), there must be a pit at (2,0) Since the two cannot occupy the same room.

- After picking up the gold, the display looks like this:

```
Aldrac Dungeon Simulator
Avoid at all costs the pits and the evil Aldrac, cursed be his name.
-----------------------------------------
|    |S   |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------
|SB  |    |SB  |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------
|    |    |    | B  |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------
|    |    | B  |    |    |    |    |    |
|    |    | U  |    |    |    |    |    |
-----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
-----------------------------------------

You are holding the gold
Enter Your move
(R)ight, (L)eft, (U)p, (D)own, (P)ickup gold:
```

  - Notice that the grid no longer shows 'G' because the user has picked it up.
  - The interface also shows "You are holding the gold" for the rest of the simulation.

- When the user returns to (0,0) while holding the gold, as in the above example, then the user escapes with it. Here is an example of that:

```
Aldrac Dungeon Simulator
Avoid at all costs the pits and the evil Aldrac, cursed be his name.
----------------------------------------
|    |S   |    |    |    |    |    |    |
| U  |    |    |    |    |    |    |    |
----------------------------------------
|SB  |    |SB  |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------
|    |    |    | B  |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------
|    |    | B  |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------
|    |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |
----------------------------------------

You've escaped with the gold, well done!
```

- Your code must compile and run.
- You code, as always, must be highly modularized.
- You must validate user input. The program **must not end** if the user enters an invalid option. You must prompt the user again if an invalid option is entered.
- The program should end if the user dies or escapes with the gold. The program will not terminate for any other reason.
- **Do not change the user interface or the behavior outlined in these specifications.**

Hints

- It will help if spend time designing the code before you write it.
- Try to implement the dungeon as a class.
- Use all that you have learned thus far.
- You will find that enums will be helpful.

Reminder

You are responsible to do your own work. **This is not a team project**. **Do not show anyone your code and do not look at, or copy, any code from any source (even the lecture notes or the book); create your own code.** Our lectures and the book contain all the information you need to complete this project. Any violation of the school's academic integrity policy or the policy of this class will result in a zero grade and an academic misconduct report filed with the school; no excuse will be accepted.

Your program must compile and run to receive any credit. If I cannot compile your program, you will receive a zero for your score. Treat this like any other exam, start right away and put effort into it.

Rubric

This project/exam is worth 200 points. The points will be distributed according to the chart below. If the program does not compile or is missing, then no points will be awarded.

Partial credit will be awarded for the features of the program that do work if the program compiles and runs.

Program	**200**	
Requirements	**60**	**(you have met the requirements of the project, especially modularization)**
Correctness	**100**	**(your program functions as expected)**
Code Quality	**40**	**(Your code is well formatted and organized—modularization is part of this too)**
Total.....	**200**	

Note: code quality refers to following good coding practices: Indentation, spacing, proper naming of identifiers, and modularization make up this part of the grade.

What to Submit

- Your source code file, *odiums_mansion.cpp*, the header file, *odiums_mansion.h*, and any other header or source files that you created.

Hint: the sooner you get started, the more you will enjoy this project, and the better you will do.