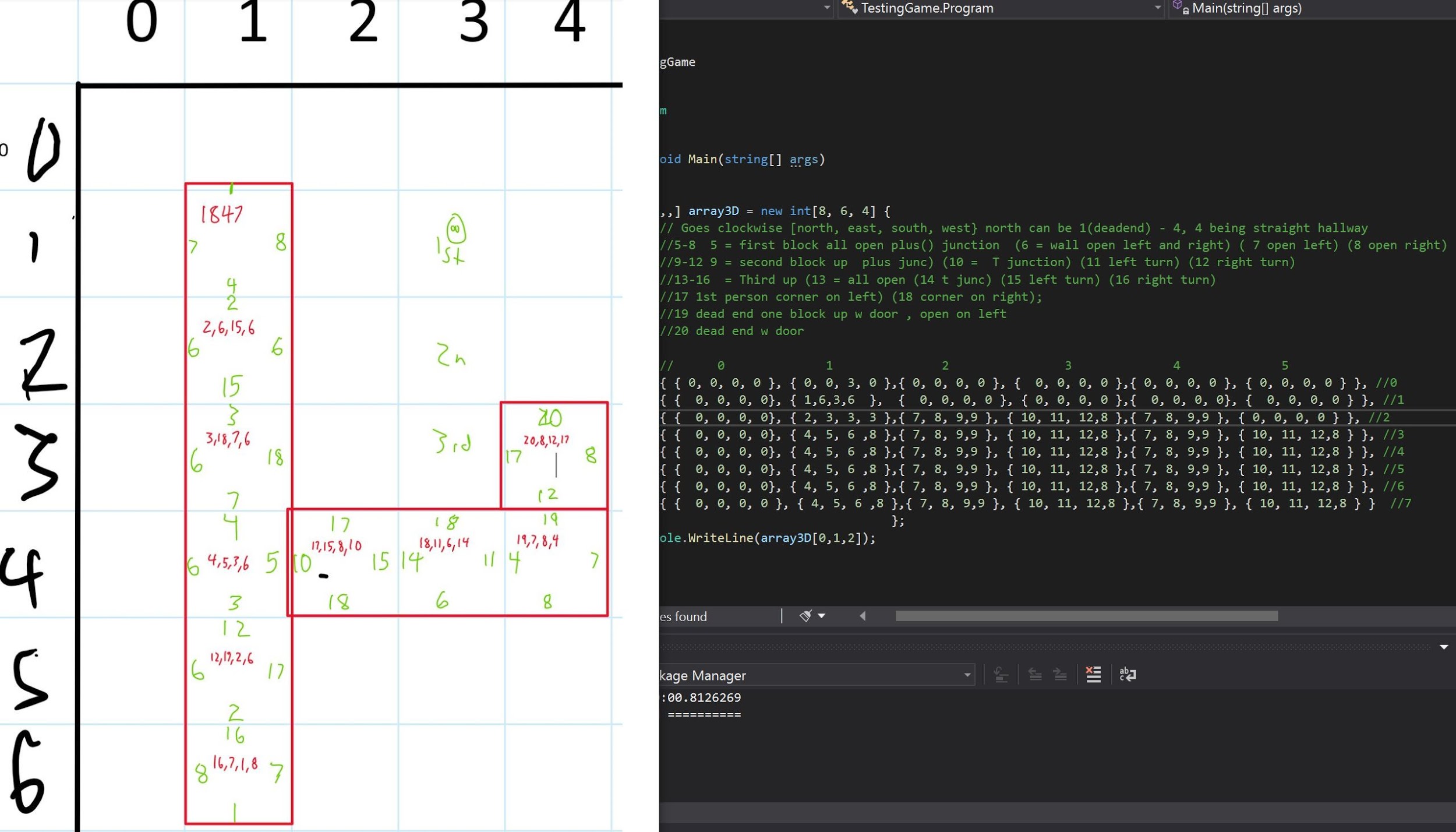
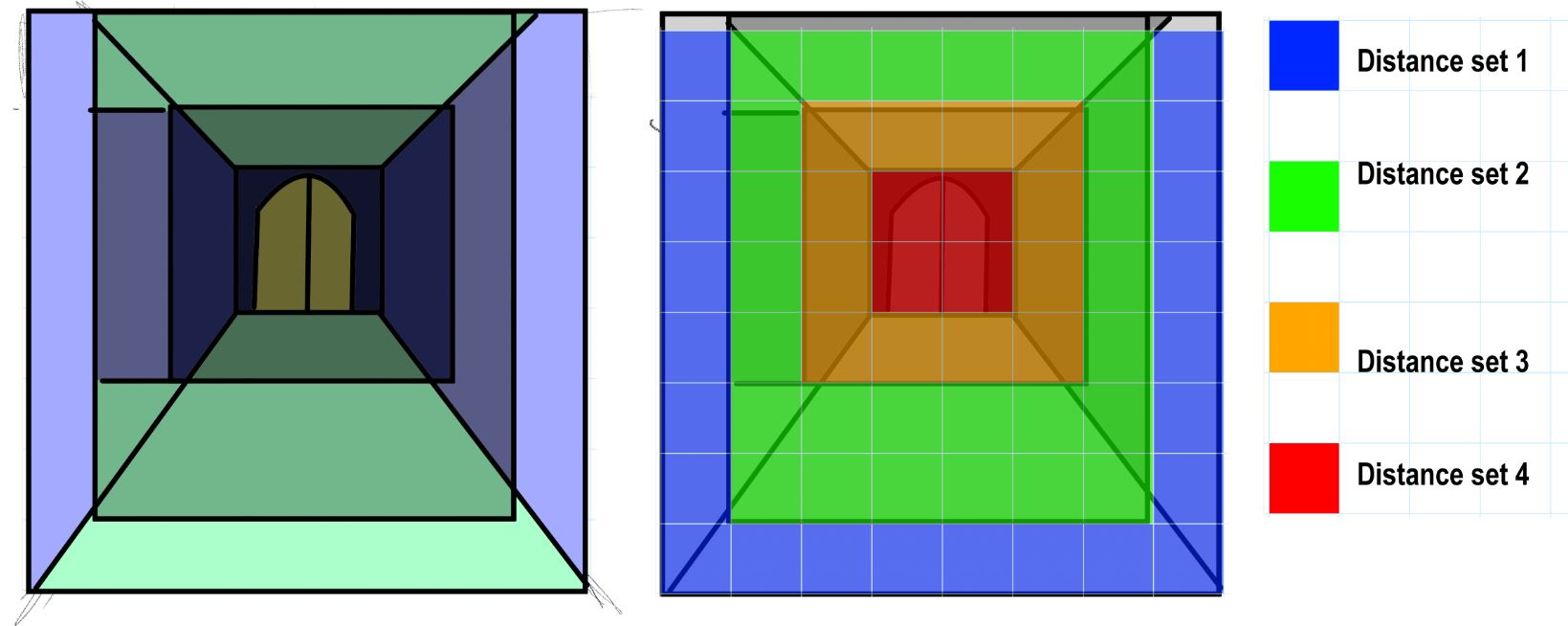
SDEV 260 Final project proposal -

* *For your final project, you will create a program that you find interesting.*

I find video games interesting! I wish to make a simple dungeon crawler. The basics will include a matrix based dungeon, a few simple tasks to complete, and graphics to make it more pleasing. I am leaving the idea open for levels of completion. At the bare minimum, all of the requisites of the final project will be met and the “game” will just be an arrow moving around a matrix. At best, there will be graphics that show what the user would be seeing. I plan on using a poly dimension array to changes with user direction-facing, x,y, and so on. If the user is facing a wall, he will see a wall etc.

I began setting up the array that will hold the location data. Each number in {0,0,0,0} represent the characters direction facing. north, east, south and west facing. Each one of these stored numbers (0- 20) represent an image combination.





In this particular level, the user must guide their hero through the dungeon, beat the secret password out some bad guy, and enter the secret password to exit the dungeon.

An array will hold the x,y, location in the matrix and will house several addition dimensions to store things like nearest wall, nearest left, nearest right, and if that location has an object of interest.

Using separate classes, I can accomplish the basic framework and use inheritance to add bells and whistles if time permits.

* *A well-developed and intuitive user interface*

I plan on using a GUI interface with buttons and images that are very user friendly.

* *Exception and Error Handling*

If the user tries to move in a direction that is not possible, exception handling will occur. Also, exception handling will occur when user data is input. The “password” must be a number.

* *Use proper OO programming technique (encapsulation, properties, and methods, inheritance (if appropriate))*

There will be many various uses of encapsulation, properties, methods and inheritance. The “location of interest”s will be a nested class.

* *Use proper resource management and garbage collection*

I had to look up resource management to understand better. I’ll be sure to practice good resource management. We haven’t used Garbage collection too much, but it seems to be based around GC. resources. I’ll include proper Garbage collection.

* *Implement interfaces and at least one abstract class*

I found this:

“The purpose of an abstract class is to provide a common definition of a base class that multiple derived classes can share. For example, a class library may define an abstract class that is used as a parameter to many of its functions, and require programmers using that library to provide their own implementation of the class by creating a derived class.”

<https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/classes-and-structs/abstract-and-sealed-classes-and-class-members>

I can’t quite understand this concept yet, but just like “properties”, I will catch on once I have had some practice implementing it.

* *Thorough commenting and a discussion of how the program was tested to ensure it is working properly.*

There will be adequate commentation.

* *A discussion of any known bugs and any instructions that you think your instructor will need when grading your project.*

I will provide ample information.

Here is my Class diagram thus far.

