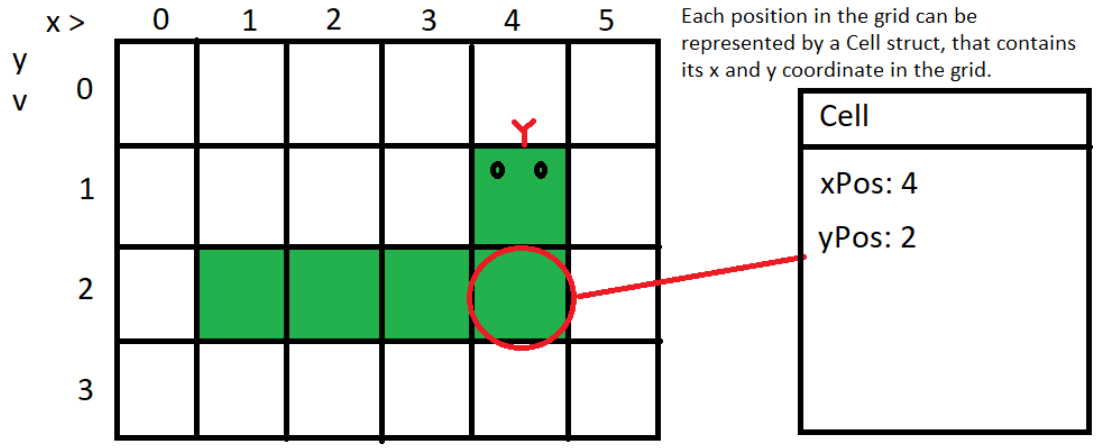
Due Date: 11/4/24 @ 11:59PM

CSC240 Assignment #9

This week we will be learning a new data structure, DS_Queues! Queues are a great data structure for when we want to maintain the order of our elements and typically only need to access each of the elements in the order they were inserted. Queues follow a First In, First Out (FIFO) order, meaning the first element inserted is the first element that can be removed (think of a line at the grocery store). Queues essentially have two main operations: enqueue(), which places an element into the queue, and dequeue(), which removes the first element in the queue from the queue and returns said element. Other operations exist within queues for checking the number of elements and whether or not the queue is empty, but these are the two main methods utilized.

This week, we are going to utilize DS_Queues to implement a classic, the game of Snake! By using a Queue and some of the structs we utilized last time for Battleship, we can represent the Snake, like so:



We can represent our Snake by using a Queue of Cells, thus maintaining their proper order once we begin moving. Notice, the Head of the Snake is stored at the Tail of the Queue and the Tail of the Snake is stored at the Head of the Queue. This is because we will want to pop off the tail for each cell that we move in the grid. A dequeue() operation will always remove the Head of the Queue. Then we can simply create a new Cell for the Snake's new Head position and insert it into the Queue!

Back of Queue (aka Tail of Queue)

*Not to be confused with the tail of the Snake

Cell	Cell	Cell	Cell	Cell
xPos: 4 yPos: 1	xPos: 4 yPos: 2	xPos: 3 yPos: 2	xPos: 2 yPos: 2	xPos: 1 yPos: 2

Front of Queue (aka Head of Queue)

*Not to be confused with head of the Snake

For this assignment, just like last time, I have created the UI, Gamemaker Objects, and defined a few Structs that are used in the implementation. You are responsible for creating the Snake struct, as it will utilize a Queue of Cells as in the diagram. The script file containing the structure of the Snake Struct is already created for you. Inside of the Snake Struct, you will find that I have provided for you the following:

- All attributes for the Snake Struct as well as brief descriptions of what each of these attributes are used to represent/control
- The names and parameters of all of the methods that you must implement. (NOTE: function names must remain the same, as they are referenced in various areas of the code)
- Documentation for each function that specifies what that function is to do, what the parameters represent, and what the return value should represent.
- One singular fully-completed .toString() method for your testing.

Due Date: 11/4/24 @ 11:59PM

You are simply responsible for filling in the empty Struct methods with the code that would perform the specified actions. I would suggest porting all of the Structs (mainly the Cell, Board, and Snake structs) out into a blank project and testing each of the functions as you write them. Make use of the .toString() methods for the structs! They will be extremely useful in making sure you are performing these actions appropriately. Once you believe that you have the Snake Struct up and running, paste this into the fully-working project that I have provided to you and if it works properly, then congratulations, you've done it!

Please note: You only need to edit the Snake Struct script file. There does not need to be any other changes made to any of the other files or code in the Gamemaker objects, however, I would highly recommend taking a look at these areas and understanding what they do, as this may make your implementation a bit easier. I have made sure to incorporate thorough documentation to guide you through all areas of the program. 😊

When you have completed the assignment, please zip up your Game folder and submit it to the Assignment 9 Dropbox.