**PROJECT DESCRIPTION**

**Game Name: Snake Frenzy**

**Category D: Animation Game**

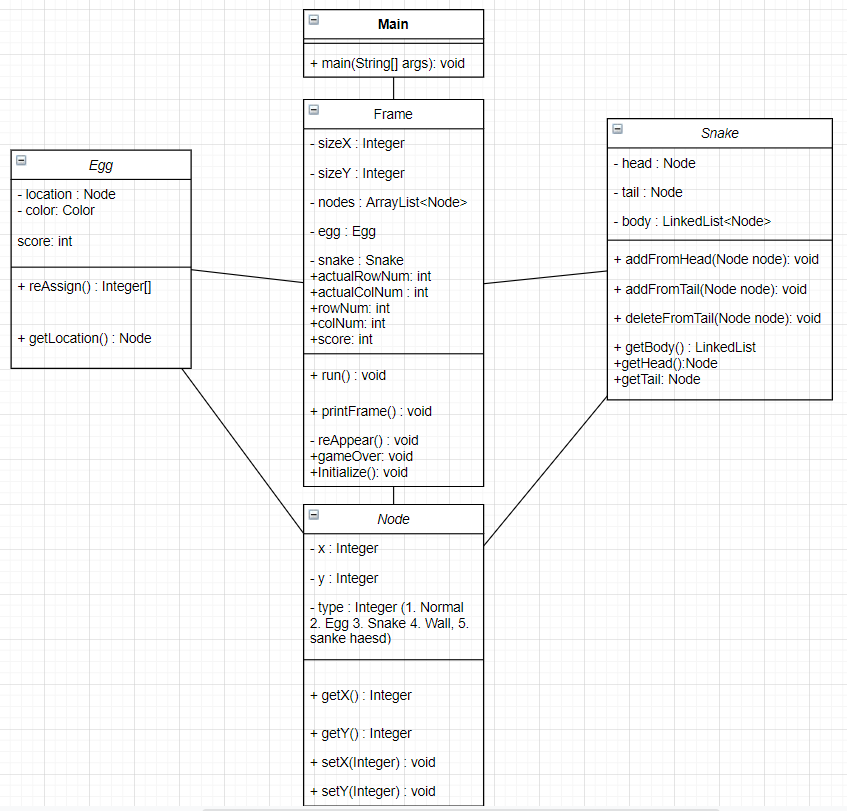
**Description:**

* Animation application game style where the snake continually moves in the game frame
* Players control the snake’s movement by pressing four keys UP, DOWN, LEFT and RIGHT on the computer to move the snake up, down, left or right, SPACE bar to either pause or resume the game
* Players try to lead the snake to eating more eggs and growing in length
* Obstacles to avoid while moving are walls and the snake’s body, and random bombs appearing after the snake eats an egg
* The game is over when the snake hits the wall (game outer frame) or its own body, or when the snake’s length is below zero due to hitting bombs
* The game has three different levels: easy, medium and difficult, each varying in speed of the snake movement at the start of the game. The easier the level, the slower the snake and vice versa. The snake’s speed will increase as the snake grows longer as well
* The score and levels will continually be updated throughout the game as the snake eats eggs

**First iteration: Text-based version of SNAKE FRENZY \_ Date: July 23, 2020**

**Class Responsibility:**

* Frame Class: Johnny
* Node Class: Ha Do
* Snake Class: Musaab
* Egg Class: Yuzhe Zhou
* Main: Yuzhe Zhou
* Obstacles: Yuzhe Zhou



**Main:**

The enter path of the frame.

**Frame**

1. actualRowNum: width of the feild where snake can move.
2. actualColNum: height of the feild where snake can move.
3. rowNum: width of the interface.
4. colNum: height of the interface.
5. Nodes: to save different type node in Node class
6. List<Node> nodes: every single node in the map in a list.
7. Egg: invoke the variable a and method in Egg class.
8. snake: invoke the variable a and method in snake class.

run method

To make the game run.

PrintFrame method

To make the node array list as the grid into the frame and reprint the frame each time if an egg appears and snake moving.

reappear method

Judge whether egg and snake are coincided, if they are, reassign the egg and add a node at the end of snake.

Initialize:

When the snake shows in the map, replace the normal node to snake node (which are snake body and snake head), also when the egg randomly shows in the map, replace the normal node to egg node.

GameOver:

Whether the head node have same position to the body, or whether the head node out of the map, or the head touch the obstacles.

**Node**

x and y: the position of the node.

type: node has five types, 1 is normal grid, 2 is egg, 3 is snake, 4 is wall, 5 is snakehead

**Snake:**

AddFromHead method:

To add the node in the head.

DeleteFromTail method:

To remove the last node to the head

AddFromTail:

Add a node from tail

**Egg:**

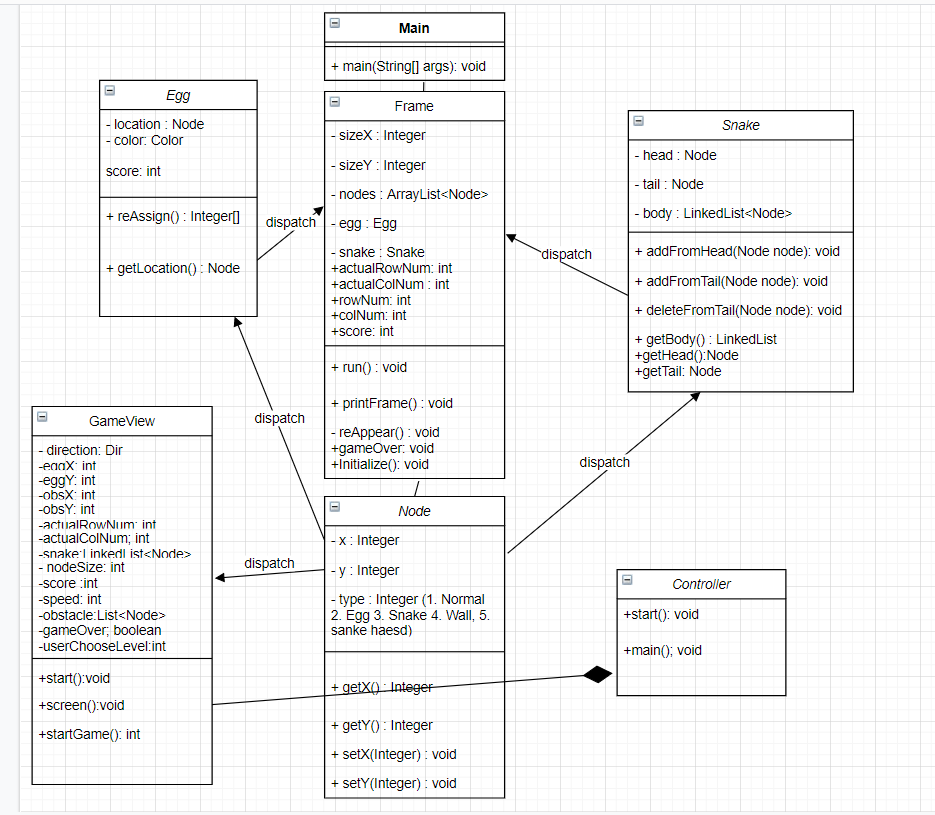
Color: Set a color for the egg.

Location: the egg in a specific coordinate of x and y

Reassign:

Randomly assign the egg in the frame.

**Second iteration: GUI version of SNAKE FRENZY \_ Date: August 6, 2020**



**Class responsibility:**

GameView: Zonglin Zhang, Ha Do

Controller: Yuzhe Zhou

**GameView:**

direction: this enum is giving the game direction: left, right, up, down.

EggX and eggY: give a random location for the egg.

ObsX and obsY: give a random location for the obstacles.

ActualRowNum and actualColNum: the area that the snake can move.

Snake: linkedlist<Node> when the snake ate one egg, the list increase one size.

Nodesize: give the size for each node.

Score: start with 0, after snake egg one egg, score plus one.

Speed: the initialize the speed of the snake. This is used to construct different levels

Obstacle: arraylist<Node> when the snake ate one egg, the list increase one size, and show a new random obstacle in the map.

GameOver: to judge whether this game over or not, if true game over, if false continue.

UserChooseLevel: the level index that user choose.

Start( ): this method is to set the map and make the game run.

StartGame( ): make sure the egg and obstacles are not shows in the same location with snake body.

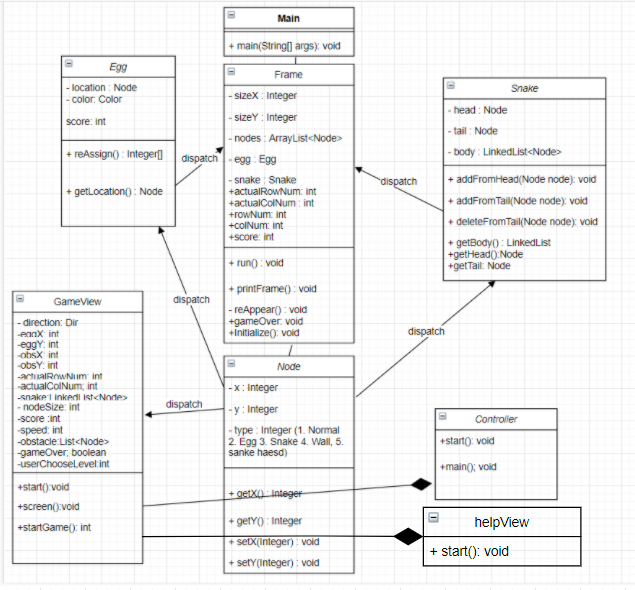
Screen (): To show the snake and make it move in the screen and show the egg and obstacles.

**Controller:**

start (): this part is to set different buttons and labels for the first scene and when we click the start button the scene will switch.

Main (): this method this to launch the whole game.

**Final iteration: GUI version of SNAKE FRENZY \_ Date: August 11, 2020**



**Class responsibility:**

helpView: Yuzhe Zhou

**helpView:**

start(): this method creates a scene to provide text instruction for players on how to play the game