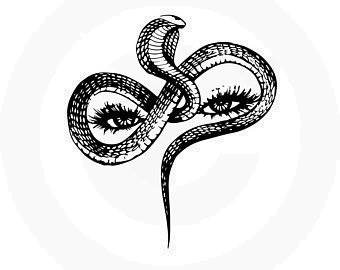
CS 106 Project REPORT

“Snake Eyes”



**Oni Luca**

01.04.2021

CS 106

# INTRODUCTION

Hello, my name is Oni Luca and this is my project report for the “Snake Eyes” program.

Before I start my report here is a brief description of what is required.

**Part A**-Create an application that simulates playing a dice game called “Snake Eyes”. In this game, a player rolls two dice. Each die has six faces. Each face contains 1, 2, 3, 4, 5, or 6 spots. After the dice have come to rest, the sum of the spots on the two top faces is calculated. If the sum is 7 or 11 on the first roll, the player wins. If the sum is 2, 3, or 12 on the first roll the player loses. If the sum is 4, 5, 6, 8, 9, or 10 on the first roll, that sum becomes the player’s “point”. To win, a player must continue rolling the dice until the player rolls the point value. The player loses by rolling a 7 before rolling the point.

**Part B-**Enhance the game so that multiple users can play simultaneously. Have an option for the computer to play as well. The program should ask how many players participate if the computer will play and the total number of rounds for the game. Record the scores of the players on a scoreboard. Points are counted as follows:

• +10 points if you win with the first roll

• +(10-x) points, if the player wins after x rolls

• -5 points, if the player loses on the first roll

• -3 points, if the player loses on any other roll

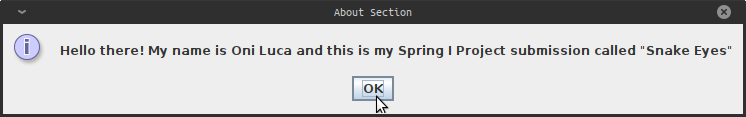
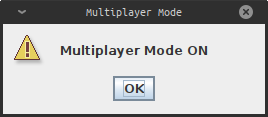
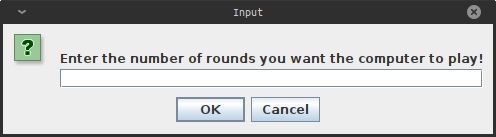
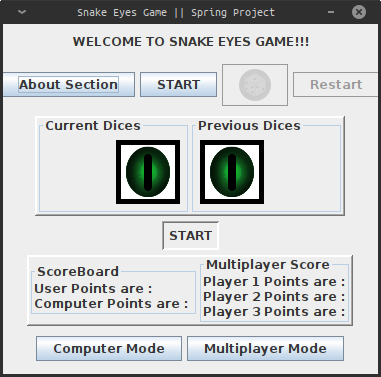
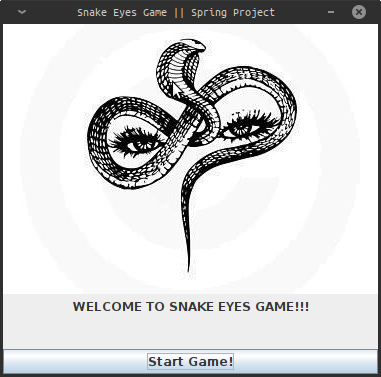
# Design

For the design of this project, I have work on something we might have touched a bit on class but not that much and that would be Layouts. For this project, I have used a myriad of layouts such as CardLayout which is used to hold a starting pane and the main program pane. FlowLayout is used in the main program and there is also BoxLayout and BorderLayout which are used in specific panels. The starting pane holds an Image Lable, the project title, and a button that is used to start the game and also switch panes.

The main program begins with the title of the game and then a row of buttons. The row contains the Start button, Roll-Again, Restart, and an about section button. After the row of buttons, we have two titled borders panels inside a beveled border panel that contain the image labels of the currently rolled dices and previously rolled dices. The dices are designed on photoshop by myself and the roll again button icon is found on a royalty-free page. Which are followed by a label message which instructs the user on what to do.

Following are two scoreboards: the user and computer scoreboard and a multiplayer scoreboard. The normal user scores are changed in each game the user plays, whereas the computer score is changed when the user clicks the computer mode button down below. Next to it at the bottom is the multiplayer mode button which allows up to three users to play a round each.

This would conclude the design portion. Below the design is displayed.



# Code

1. Constructor

In the beginning, I set the title of the frame and color. After that, I declare all the panels I will use whether that would be CardLayout panes, button holders, or scoreboard panels. After creating the CardLayout Panes I set the size of them and color. Then I declare each image icon I used such as dices, button, etc. Labels come next, such as titles, messages, and scores. Then 7 buttons are created: StartGame(used in the first pane), About Section, Start, Roll-again(an image is set instead of a label), Restart, Computer-Mode, and MultiPlayer-Mode.

(I have also used hover pop-ups in some labels and button for the user to find when having difficulty)

After creating everything I add them to each corresponding panel and set borders and layouts for them.

In the end, the action handler is created and each corresponding action listener.

1. Action events

* StartGame- Switches panes and starts the main program.
* Start- Generates two random dices and sets an image based on the number rolled. There are if statements whether the user wins or loses. When the user neither wins nor loses the dices generated are set as previous dices and their sum is set as result and the button disables itself.
* Roll Again- in the second roll of the game the dices are re-generated. If statements compare the new sum to 7 in case it loses or the result to win. In both cases, the button disables itself and enables the restart button. The third case would just increment a roll-counter so the user can roll again until it wins or loses.
* Restart- The restart button when clicked will disable itself and enable the start button. It will also set the new points which are counted in the start and reroll buttons to the mode the user is in whether that would be a normal mode where user points are set, computer mode where computer points are changed and multiplayer mode where the three users points are changed one at the time. The mode is detected with if statements that are triggered when computer mode or multiplayer mode set a boolean variable to true.
* Computer-Mode- This button firstly pops up an input dialog box that asks the user for a number of rounds. The input is used as a loop counter that makes the game autoplay itself for the number of rounds given. DoClick is used for the game to simulate a computer playing. It also sets a boolean variable to true which makes sure the points won or lost in these rounds are awarded to the computer.
* Multiplayer-Mode- Firstly it pops up a warning box and sets the boolean variable to true. For three rounds the points are awarded to three different users and then the game continues to play in normal mode.

# Diagram of inter-dependencies for the classes-

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Gui()** |  |  |  |
| **Variables** |  |  |  |
| **Name** | **Description** | **Type** | **Initial value** |
| Dice1/2 | Random numbers generated | Integer | 0 |
| Sum | Sum of two random dices | Integer | 0 |
| Result | If player doesnt win or lose first round> sum=result | Integer | 0 |
| Roll number count | iIf player has to reroll, counts numer of times reroll | Integer | 1 |
| Multiplayer count | Number of users playing in multiplayer mode | integer | 3 |
| Point won | Points won by user | Integer | 0 |
| Computer points | Points won by computer | Integer | 0 |
| Multi user point | Points won by user in multiplayer mode | Integer | 0 |
| User 1 points | Points of user derived from multi user point | Integer | 0 |
| User 2 points | Points of user derived from multi user point | Integer | 0 |
| User 3 points | Points of user derived from multi user point | Integer | 0 |
| Computer Check | Checks if computer is playing | Boolean | N/A |
| Multiplayer check | Checks if multiple users areplaying | Boolean | N/A |
| Project title | Name of the game used starting page | JLabel | N/A |
| Game name | Name of game used program page | JLabel | N/A |
| Current dice 1 | Holds image of first dice | JLabel | N/A |
| Current dice 2 | Holds image of second dice | JLabel | N/A |
| Previous dice 1 | Holds image of first roll first dice | JLabel | N/A |
| previous dice 2 | Holds image of first roll second dice | JLabel | N/A |
| message | Displays messages for user | JLabel | N/A |
| user score | Displays user score | JLabel | N/A |
| computer score | Displays computer score | JLabel | N/A |
| user 1 score | Displays user1 score | JLabel | N/A |
| user 2 score | Displays user2 score | JLabel | N/A |
| user 3 score | Displays user3 score | JLabel | N/A |
| about section | Displays author info | JButton | N/A |
| start | starts game | JButton | N/A |
| roll again | roll again in case user neither wins or loses | JButton | N/A |
| restart | sets points of user based on modes | JButton | N/A |
| starting button | changes card panes | JButton | N/A |
| computer mode | Sets game on computer mode | JButton | N/A |
| multiplayer mode | sets game on multiplayer mode | JButton | N/A |
| starting panel | contains starting header and starting button | JPanel | N/A |
| starting header | contain game tittle | JPanel | N/A |
| program panel | contain all the panels below | JPanel | N/A |
| program header | contain game tittle | JPanel | N/A |
| buttons | contain all buttons | JPanel | N/A |
| current dices | contain current dices label | JPanel | N/A |
| previous dices | contain previous dices label | JPanel | N/A |
| messages | contain message label | JPanel | N/A |
| scoreboard | contains user and computer points | JPanel | N/A |
| multiplayer scoreboard | contain all user points | JPanel | N/A |
| scoreboards holder | holds previous two panels | JPanel | N/A |
| diceholder | contains current and previous dices panels | JPanel | N/A |
| computer multiplayer buttons | contains computer and multiplayer mode buttons | JPanel | N/A |
| card | Layout manager holder | CardLayout | N/A |
| cardPane | holds two card panes panels | JPanel | N/A |
| **Methods** |  |  |  |
| **Name** | **Description** | **Parameters** | **Return Type** |
| actionPerformed | Here all actions are performed | ActionEvent | void |
| setTitle | Sets Title of the game | String | void |
| setDefaultCloseOperation | Method to close window | int | void |
| setPreferredSize | Sets size of a Jcomponent | Dimension | void |
| setFont | Sets the font | Font | void |
| setBackground | Sets background | Color | void |
| setEditable | Sets Jcomponent editable or non-editable | boolean | void |
| setTitleFont | Sets the font for the title | Font | void |
| setBorder | Sets the border | Border | void |
| setVisible | Sets Jcomponent visible or not | boolean | void |
| setLayout | Sets layout manager | LayoutManager | void |
| addActionListener | Adds ActionListener to button | ActionListener | void |
| setActionCommand | Sets action command for button | String | void |
| add | Adds a Jcomponent | Component | Component |
| repaint | Repaints component | N/A | void |
| setResizable | Sets frame resizable or not | boolean | void |
| setText | Sets text | String | void |
| setEnable | Enables or disables component | N/A | void |
| Math.Random | creates random integer | int | int |

# 

# 