**CIS 162 Lab 7**

## A Simple Dice Game

**Objectives**

After completing this lab, you should be able to:

* *write* conditional statements
* *write* programs that include the use of more than one class.
* *Use a GUI class*

**Step 1: Create a New BlueJ Project – Lab7**

**Step 2: Class Definition: GVdie**

Download the GVdie class and use the <edit> <Add Class from File> from BlueJ to include the GVdie class in the Lab7 project created in step 1. It should compile with no errors.

You will not need to make any changes to the code of the GVdie class but will need to understand how to use each of the following methods:

GVdie d1; // Declare d1 as a GVdie object

d1 = new GVdie(); // instantiate the GVdie object

d1.roll(); // roll the die d1

v1 = d1.getValue(); // check current value of d1

d1.setBlank(); // set face of d1 to blank

**Step 3: DiceGame class**

* Create a new class called DiceGame.
* **Instance variables**: Provide appropriate names and data types for each of the instance variables.
  + three GVdie objects. Example: private GVdie d1;
  + credits (int)
  + guess (int)
  + current value for each die (int)
  + message (String).
* public DiceGame () - a constructor that initializes all of the instance variables to appropriate values:
  + *instantiates* three dice

//example to instantiate d1 as a GVdie object

d1 = new GVdie();

* + sets credits to 100
  + sets message to "Welcome to my Game"
  + sets the dice to blank

//example to set a die to blank

d1.setBlank();

**Accessor methods**

* public String getMessage () – return the message
* public int getCredits () – return the credits
* public GVdie getDie (int num) - return the requested die. Legal values for the parameter are 1 - 3. The GUI uses this method to show the graphical representation of the dice.

Example: switch (num) {

case 1:

return d1;

**Mutator methods**

* public void setGuess (int g) – set guess to g.
* public void restart ()
  + sets credits to 100
  + sets message to "Welcome to my Game"
  + sets the dice to blank
* public void playGame ()
  + roll the three dice invoking the private method rollDice();
  + compare the guess with the new values by invoking your private methods below: isTriplet(), isDoubles(),wasNumberRolled( )

**NOTE: You may want to write this method when you finish writing the private methods found below.**

if (isTriplets()){

credits += 50;

message = "Three of a Kind!";

}

else if (...

Update the credits based on the game rules:

|  |  |  |
| --- | --- | --- |
| **matches** | **credits** | **message** |
| Three matches of the guessed number | adds 50 to credits | "Three of a Kind!" |
| Two matches of the guessed number | adds 20 to credits | "A pair!" |
| One match of the guessed number | adds 10 to credits | "Only one match! |
| No match of the guessed number | reduce credits by 10 | "No match - you lose! |

Update the message according to the table. There are not System.out.println statements in this class.

**Private helper methods.**

* private void rollDice ( )
  + roll the three dice by invoking each roll method:

Example: d1.roll();

* + update the value associated with each die

Example: v1 = d1.getValue();

* private boolean wasNumberRolled( )- return true if the guess was rolled, return false otherwise.
* private boolean isDoubles ( ) – return true if the guess is rolled twice, return false otherwise.
* private boolean isTriplet ( ) – return true if all three dice match the guess, return false otherwise.

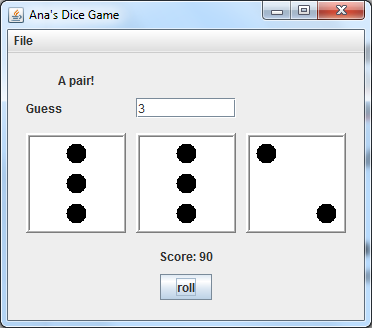
**Step 7: Class Definition: DiceGameGUI**

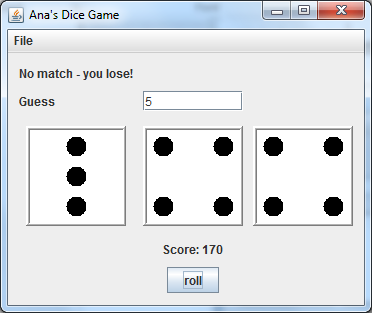
Download the DiceGameGUI class and use the <edit> <Add Class from File> from BlueJ to include the DiceGameGUI class in the Lab7 project created in step 1. It should compile with no errors.

**Note:** If it does not compile, you may have changed the names and/or headers of the methods in the DiceGame class.

You do not need to make any changes to this class. Read the code of this class to understand how the GUI works.

Run the main method of this class





**Grading Criteria**

This lab is worth a possible 10 points. Show your work to the instructor or lab assistant.