CISC 110 Lab 5: Die-Rolling Simulation

In this lab, you will complete the first portion of Assignment 5. In your assignment, you will create an app that simulates throwing a six-sided die (only one die, rather than a pair of dice) many times in order to do a statistical analysis of how random the simulation actually is.

The user chooses the number of trials and specifies a particular die value (1,2,3,4,5, or 6). During each trial, the app will simulate rolling the die until the user's specified die value is rolled. After completing all of the trials, it calculates the average number of rolls it took to roll the user's die value over all of the trials. Since there are six numbers on the die, if there is an even probability of rolling any one of them, it should take an average of six rolls to roll a desired value. The user can also specify weights for each side of the die, so that the probability of it landing on one side will be greater than another. Your simulation will also include an animation of the die rolling and stopping on a random side.

In the lab portion, you will write the code to run the trial with sample values, instead of with input from the user, and will display the results only via trace statements. As well, the weights and the die animation won't be included. For a similar example, look at the Wk5WhileLoops examples on the CISC 110 web page in the Schedule table under *ActionScript and Flash Examples*.

Here are the required functions:

Function Name	Description
`Assign5`	The main function as well as the class constructor. This function is responsible for adding a listener for a Button click and creating a custom roll value for the die, which it initializes to 0 (used to represent no roll).
`runTrial`	The event handler for when the Button is pressed. This function runs the simulation of dice rolls by calling the rollDie function many times.
`rollDie`	This function simulates rolling the die once. It sets the custom roll value of the die to 1,2,3,4,5, or 6.

Here are the required display objects:

Object Identifier	Description	Type of Object
`die`	MovieClip instance: This is the die that is rolled.	MovieClip
`startBtn`	Button instance: When pressed, starts the die rolling simulation	Button

Here is the required custom property for the MovieClip instance:

MovieClip Instance	Description
`die`	`rollVal`: the current roll value of the die (1,2,3,4,5, or 6)

Here are the steps for you to complete:

1. Create a new ActionScript 3.0 file called `Assign5.fla` and save it in an `Assign5` folder. Create your display objects (Steps 2 & 3).

2. Insert a new MovieClip symbol and create a simple die, which you can improve upon later, for instance just a square with a number 1 on it. Drag an instance of it onto the stage and give it the instance name `die`.

3. Insert a new Button symbol and create a simple button, which you can improve upon later. Drag an instance of it onto the stage and give it the instance name `startBtn`.

4. Specify your document class to be `Assign5`. Next write your code (Steps 5-8).

5. Download the file `Assign5.as` from the CISC 110 website under Lab 5. Then make the changes below. **Remember to save back-up files** under other file names. Then you can always go back to one of those files, if needed, open it and rename it `Assign5`. If you are using your own laptop, you also need to keep back-up files elsewhere, for instance on a memory stick, in case you have computer problems. If you do not do this, you will have to recreate your assignment if you have problems.

6. Within your constructor function, `Assign5`, complete the tasks specified in the comments:
   // TASK 1: Create a custom `rollVal` property for the ship and initialize it to 0
   // TASK 2: Add a button event listener to allow the user to start the trial

7. Write another separate function, called `rollDie`, that randomly generates an integer between 1 and 6 and sets that to be the value of die's roll value.

8. Within your button handler function, `startTrial`, complete the tasks specified in the comments:

```
// TASK 1:  Define the following variables and give them sample values:
//   - A variable for the total number of trials to be run, for instance 100 or 1000
//   - A variable for the user's specified roll value: 1,2,3,4,5, or 6
//   - A variable used to count the total number of rolls of the die for all of the trials
//     (several rolls per trial), initialized to 0

// TASK 2:  Set die's rollVal to 0.  Then, with a while loop, roll the die
//      by calling rollDie() until the user's specified roll value is rolled.
//      At each roll, add 1 to the count of the total number of rolls and display the
//          roll value with a trace statement
```

Here is the call to rollDie( ) that needs to be inside your while loop:

```
      rollDie( );
```

```
// TASK 3: Repeat Task 2 for each trial via a while loop that contains both the
// statement that sets die's rollVal to 0 and the while loop created for Task 2

// TASK 4: With trace statements, display the number of trials
// and the total number of die rolls (several per trial)

// TASK 5:  Calculate and display the average number of rolls that were required
//  to obtain the specified roll value by dividing the total number of rolls
// by the number of trials
```

Lab 5 Marking Scheme (1% of final mark)

Marked out of 5:

1 mark - A MovieClip, called `die`, with a custom property called `rollVal`, and a Button called `startBtn` created.

1 mark - A function defined, called `rollDie`, that randomly generates an integer between 1 and 6 and sets that to be the value of die's roll value.

1 mark - A while loop within the Button handler that rolls the die by calling rollDie() until the user's specified roll value is rolled.

1 mark - Repeats Task 2 for each trial via a while loop that contains the while loop created for Task 2

1 mark - Calculates and displays the average number of rolls that were required to obtain the specified roll value