**Computer Programming I**

**Prof. Tom Wulf**

**Ass 02 Craps Game**

**Fall 2019-20**

**20 Pts (No graduate or Extra credit)**

**Learning Goals:**

* **Conditional Structures in Java if, if else, cascaded if, nested if, while, do..while**
* **Random numbers in Java for simulations. Gaussian vs Linear Probability**
* **Testing. Make sure your programs run by testing them. Correct them if they do not.**
* **Important: later on, I will ask you to create a simulation program that compares using 2 6-sided die with Gaussian probability vs a single 10 sided die with linear probability.**
* **Students often ask “Why craps?” well back in the day, everyone knew the rules for this simple dice game. We use it because it is a problem that includes an indefinite number of repetitions (i.e. rolling against the point) which you have to code with a regular while loop.**

**Mini – Lecture: Random in Java**

import java.util.Random; // must import Random (like we do Scanner)

Random rnd = new Random(); // create a Random object called rnd.

int rndBirthMonthOff = rnd.nextInt(12); // generates a random value from 0 – 11

int rndBirthMonth = rnd.nextInt(12) + 1; // What we want: 1 -12 so we shift it by adding 1  
  
int dieOff = rnd.nextInt(6); // generates a random int between 0 and 5

int die = rnd.nextInt(6) + 1; // generates a random int between 1 and 6 **what we want for dice**

int die1 = rnd.nextInt(6) + 1;

int die2 = rnd.nextInt(6) + 1;

int crapsRoll = die1 + die2; // Gaussian range from 2 to 12 as in rolling two dice

**Project Name: Ass02\_Craps**

1. Implement a program that simulates the game of Craps. Use IntelliJ and if directed GitHub for source control. The computer will display the rolls of the game using random numbers to generate them and will prompt the user to play again. In addition to the die result. The computer will display an interpretation of the results for each roll.  
     
   **The Rules of the Game**:
   1. There is an initial roll of 2 dice resulting in a sum of 2 – 12.
      1. If the sum is 2, 3 or 12 it is called "craps" or "crapping out" and the game is over with a loss.
      2. If the sum is 7 or 11 it is called a ‘natural’ and the game is over with a win.
      3. For all other values, the sum becomes ‘the point’ and the user makes subsequent rolls until they either throw a 7 in which case they loose or they make the point sum in which case they win.
   2. After a win or loss the next player rolls the die for a new game. In our simulation, the program will simply ask the user if they want to continue to play.
2. For each roll show the value for each die and the sum.
   1. For case i or ii above, indicate that the user either crapped out or won with a natural and then prompt to play again.
   2. For case iii indicate that the sum is now the point. For each subsequent roll indicate the status of the result:
      1. Trying for point
      2. Made point and won. (Prompt user to play again.)
      3. Got a seven and lost (Prompt user to play again.)

Graphical user interface, text

Description automatically generated

1. **EMBED SCREEN SHOTS OF NETBEANS HERE SHOWING YOUR PROGRAM RUNS: Include at least the 3 options above i .. iii**

**Submitting your work:**

Submit this file with your screen shots: **Lastname\_Firstname\_Ass\_02 Craps.docx** using your name.

If using GitHub submit a link to your repo for the project.

Otherwise, create a new compressed .zip archive folder. (Don’t give me any other type of archive, it will be returned you ungraded!) called **Lastname\_Firstname\_Ass\_02 Craps.zip** using your name and include your complete IntelliJ project.. Add a copy of this word docx file to the archive.

**Use the Canvas Assignment mechanism to submit your word docx file and either your repo link or your file archive.**