## Lake Washington Institute of Technology

## CS 143 Computer Science II

## Programming Assignment 2

## First to One Simulation

### Introduction

This programming exercise is based on Programming Challenge 11 from the Gaddis text. You are to write the simulation for a game played by two or more players. In the game, each player starts out with some number of points. Each player takes a turn rolling a dice. The amount generated by the dice is subtracted from the player’s points. The first player with exactly one point remaining wins. If a player’s remaining points minus the amount generated by the dice results in a value less than one, then the amount is added to the player’s points.

At the start of the game, prompt the user for these values:

* the number of points to start with
* the number of players

Once these values are provided by the user, the simulation runs to completion, playing itself until one of the players wins (i.e., gets a score of one). This is a simulation. It is not supposed to be a game played interactively by the user.

As the simulation proceeds, output lines of text showing the status after each player’s turn, indicating:

* what player is playing (e.g., player 0, player 1, etc.)
* what value was generated by rolling the dice
* points remaining for that player
* when one of the players wins, announce the winner

### Constraints

Your solution should include at least these classes:

* FirstToOne – this class is the driver for the simulation. It includes method main. It contains code for prompting the user for input, instantiating the players, and outputting the progress.
* Player – this class contains:
  + a constructor that takes an int parameter indicating the number of points to start
  + the current score
  + a method for a single turn (roll of the dice)
  + a getter to return the current number of points left
* Dice – provided in canvas. In the 3rd edition of the Gaddis text, the code is presented in Code Listing 6-14 (die.java).

You may include other classes that you feel are necessary for your solution. But collapsing the functionality into a smaller set of classes that the three listed above may incur a point penalty. Think object oriented.