CSCI 1320 Computer Science I: Engineering Applications – Fall 2018
Instructor: Zagrodzki
Lab 8: Due Friday, Nov 9, by 6pm
Assignment 8: Due Sunday, Nov 11, by 6pm

Objectives: implement a real application with all the challenges associated with it

- Understand the idea of modular programming in C++
- Understand different parts of the game and translate this into different modules/functions of your program
- Understand the idea of function stubs and when to use them
- Create and test functions, one at a time
- Put everything together and test everything together at the end

The top of your file should include your name, course no., assignment no., and instructor name.

Let's play Dirty Dice

The assignment's objective is to create a C++ based game in which one player (the user) competes against the computer where each player tries to steal the points from the opponent. Each player starts out with 100 points. The game is won by the first player to reach a set number of points. The number of points required to win the game is to be inputted by the user. Each turn, the player (human or computer) repeatedly rolls a dice until either a 3 is rolled or the player chooses to hold. If player chooses to hold, the sum of the rolls (i.e. the *turn total*) is calculated. At any time during a player's turn, the player is faced with a binary decision:

0. **roll** - If the player rolls a
    a. **3**: the player steals 3 points from the opponent and it becomes the opponent's turn.
    b. **1, 2, 4, 5, 6**: the number is added to the player's turn total and the player's turn continues (i.e. gets to roll again).
1. **hold** - The *turn total* is calculated, the points are subtracted from the opponent's score, and it becomes the opponent's turn.

Check whether a winner has been established at the end of every turn. The player who takes the first turn in the game is chosen randomly.

Do NOT use: global variables, arrays, pass by reference functions, or pass by pointer functions.

Requirements:

1) Your program should contain the following three functions (besides the main function, of course):

`roll` - takes in nothing. Returns a random int between 1 and 6.

`oneTurn` - takes in a single boolean variable to indicate whether it's user's or computer's turn (e.g. 1 = human turn, 0 = computer turn). Plays out the entire turn until player either rolls a 3 or chooses to hold. Returns the turn total as an integer value. *(Note: this function should be calling the* roll *function).*

`loopGame` - takes in a single integer value for game points goal (i.e. the number of points player needs to reach to win the game). Continues to repeat until either computer player or human player has won. Returns a boolean value (1 = computer win, 0 = human win). (*Note: this function should be calling the* oneTurn *function*).

2) Your program should interactively display the player and the computer choices, the dice rolls, the running turn total and the running score after each turn.

Lab 8:

For the lab assignment this week, come up with the algorithm for the project. Create a skeleton code with function stubs. Figure out what variables the functions take in, what they return, and what variables you will most likely need. You comments should contain the pseudocode describing the logic of your program. Make sure your skeleton code compiles. Submit your .cpp file on Moodle under Lab 8 by the due date.

Assignment 8:

The assignment should contain a fully functional Dirty Dice game C++ source code. Submit your .cpp file on Moodle under Assignment 8 by the due date.

NOTE: if your C++ file does not compile with standard g++ compiler in Jupyter Hub you will get a zero on the C++ portion of the assignment (no partial credit). This will be the case with all C++ assignments going forward.