

Fall 2016 CST8234 – C Programming

Lab 02: Dice Game

Setup:

1. You are to use a Linux virtual machine, Ubuntu or Fedora recommended, 64-bit or 32-bit is OK
2. Create a directory `Lastname_02`. You are going to develop your lab here!

Specifications:

You are to implement a dice game in which rolls of a pair of dice. The rules of the game are the following:

1. The player rolls the dice and adds up the face values.
2. If the first roll is a 7 or 11, the player wins.
3. If the first roll is a 2, 3 or 12, the player loses.
4. If the first roll is any other number, that sum becomes the player's point.
5. To win, the player must continue rolling the dice until he/she “makes point.”
6. The player loses by rolling a 7 before the point.

The following demonstrates a few executions of the program:

```
# ./02_Dice
                                ROLL THE DICE WITH [ENTER]
-----
ROLL NUM    DICE #1    DICE #2    TOTAL ROLL    POINT MATCH
-----
          1          2          6          8          8
          2          1          2          3          8
          3          6          2          8          8
Congratulations you roll 8 and WON
Another Game? [ Y / N ]  Y
                                ROLL THE DICE WITH [ENTER]
-----
ROLL NUM    DICE #1    DICE #2    TOTAL ROLL    POINT MATCH
-----
          1          4          3          7          7
Congratulations you roll 7 and WON at your first try!!!!
Another Game? [ Y / N ]  Y
                                ROLL THE DICE WITH [ENTER]
-----
ROLL NUM    DICE #1    DICE #2    TOTAL ROLL    POINT MATCH
-----
          1          1          1          2          2
Sorry, you roll 2 and you loose!!!
```

To terminate the program:

```
Another Game? [ Y / N ]N
Thank you for playing
You won 0 games and lost 3 games!
Better luck next time!

Another Game? [ Y / N ]N
Thank you for playing
You won 3 games and lost 0 games!
What a winner!
```

In order to successfully complete this program and obtain all the marks, you will need to:

1. Define **WON** and **LOST** as macros in your program. Use the values of 0 for **WON** and 1 for **LOSE**
2. Implement a function, with function prototype **int rollDice(void);**
 - (A) **rollDice()** should use **rand()** to randomly generate a number between 1 - 6
 - (B) return the number generated by **rand()**
3. Implement a function, with function prototype **int playGame(void);**
 - (A) When the player is ready to play, (s)he would use the key **ENTER** to roll the dice
 - (B) If the user wins in his/her first roll, congratulate the player and return with **WON**
 - (C) If the user loses in his/her first roll, congratulate the player and return with **LOSE**
 - (D) Let the user keep playing until (s)he wins / loses, give an appropriate message and finish the game with the last roll value.
4. Your **main()** should
 - (A) Call your function **playGame()**
 - (B) Ask the user if (s)he wants to continue playing another game, keeping track of the numbers of losses and wins
 - (C) **When the user decides to finish playing, display the number of wins and losses (s)he had.**
 - (D) Give the user an appropriate message depending on the number of wins or losses (s)he had
 - (D) Return with a value of **EXIT_SUCCESS**
5. Your program should use at least once the conditional operator
6. Your program should present information to the user in clear way. In the output given, a table is presented with the number of roll, the value of the first and second dice, the total of adding the dice and the points that the user needs to achieve.
7. Your program should be compiled with the flags **-Wall -ansi -pedantic**

EXTRA: (50%)

1. Modify your program so the user can place a bet. Call your new program **02_Dice_B.c**
 - (A) A user starts the game with \$10
 - (B) A user can place a bet using multiple of \$5
 - (C) If the user wins the game, it makes 3 times the bet
 - (D) If the user loses the game, it loses all the money it has bet
 - (E) You can decide if the user is able to place a bet every time it rolls the dice, or every time it plays a game.
 - (F) The money won carries over the next game
 - (G) If the bet money is 0, the user can not place any more bets.
 - (H) When the user decides to finish playing, the amount won should be displayed.