≡   CISC 121 - Introduction to Computer Science (ASO) S21   ✉ 🗩 🔔   LL

Assignments › Assignment 1: The Game of "Four, Five, Six"

Assignment 1: The Game of "Four, Five, Six"

▼ Hide Assignment Information

Instructions

Coding the Game of Four, Five, Six

Introduction: In this assignment, your task is to build a console version of the dice game "Four, Five, Six". The only requirements for the game are three players, three six-sided dice, and a whole lot of luck!

Overall Objective: "Four, Five, Six" is a gambling game where the players bet against each other during each of the rounds of gameplay. The overall objective of the game is to win the bets by rolling a better dice combination than your opponent. The game continues for 15 rounds and the player who has the most chips in their bank at the end of 15 rounds wins.

Determining the Banker: The game is played with three players. One of the players is the banker. Before the game starts, each of the players roll one die. The player that rolls the highest number becomes the *first* banker.

Example: Assume that player 1 rolls 6, player 2 rolls 2 and player 3 rolls 5. Therefore, the first banker will be player 1. Player 1 will need to make bets against each of the other two players.

The banker will continue to be the banker until another player rolls a 4, 5, 6 dice combination. The player who rolls the 4, 5, 6 becomes the banker going forward to the next round until another player rolls a 4, 5, 6.

Making Bets: The banker will make bets against each of the other players in the game at the start of each round. It is fine to either assign players a fixed number of chips at the start of the game or just proceed with each player being able to bet as many chips as they would like.

Example: Player 1 may bet 1 chip against player 2 and 1 chip against player 3.

The Outcomes: The banker will roll the three dice. Here's a listing of the possible three outcomes and the dice combinations that lead to each outcome.

Instant Win	Instant Loss	Set Point
The banker will instantly win all of the bets.	The banker will instantly lose all of the bets.	The non-pair die will set the point number.
4, 5, 6	1,2,3	Pair + 5

Pair + 6	Pair + 1	Pair + 4
3 of a kind		Pair + 3
		Pair + 2

For a set point, the non-pair die will set the point number. Each (non-banker) player will get a chance to try to beat the point number that was rolled by the banker.

If any other combination, other than the dice combinations outlined in the table above, is rolled, we call this a "fumble", and the player must roll again. For example, a roll of 6, 4 and 1 would be a fumble. There are no point penalties for a fumble - the player simply re-rolls.

Example Gameplay: Here's an example gameplay scenario to help illustrate how the game is played and how points are obtained:

Step 1: Determine the banker. Each player rolls a die.

- Player 1 rolls a 3.
- Player 2 rolls a 6   **Player 2 becomes the banker -- they have the highest roll.**
- Player 3 rolls a 4

Step 2: Player 2 (the banker) places their bets - 1 chip against Player 1 and 1 chip against Player 3.
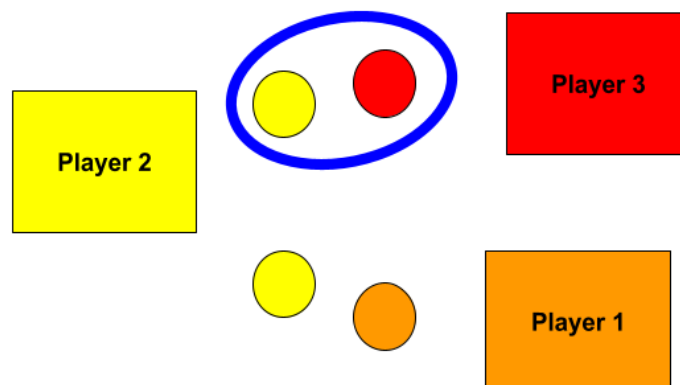
Submit   Cancel
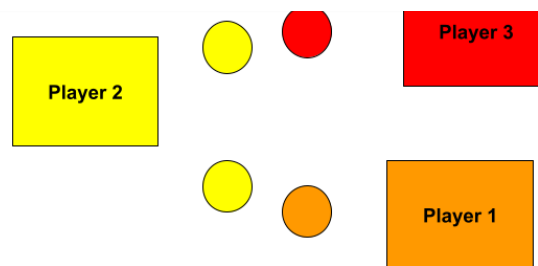
Step 4: Player 2 rolls a 4, 4, 5 -- the point is now set at 5 (a pair + 5).

**Step 5:** Player 3 rolls a fumble -- the player must re-roll.

Step 6: Player 3 rolls a 4, 4, 1 -- this is less than the point score of 5 that was set by Player 2 in Step 2.

Step 7: Player 2 collects the chips that they bet against Player 3 (which is 2 chips).

Step 8: Player 1 rolls a "three of a kind" (4,4,4) -- this is an instant win and therefore beats player 2's set point of 5.

Step 9: Player 1 collects the chips that was bet against them by Player 2.
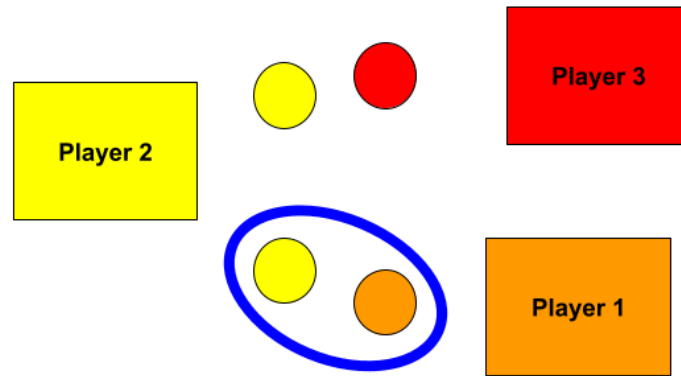
That is the gameplay for one round of "Four, Fix, Six". Another round would begin with the same banker. The banker only changes once a player other than the banker rolls a 4, 5, 6 dice combination. As soon as a player rolls a 4, 5, 6, that player becomes the banker as soon as the bets of the current round are settled.

What about a tie?

- In any given round, if two or more players tie for the highest score, there is a required reroll. Each player rerolls until they roll a pair + third die. The higher third die always wins. For example, if player 1 and player 2 are tied and player 1 rolls a 6, 6, 1 and player 2 rolls a 1, 1, 5, player 2 would win since the third die 5 > 1.
- If after 15 rounds have been completed, two or more players have the same amount of chips in their bank account (a winner cannot be determined), continue playing until a player comes out ahead and the winner can be determined.

Playing Against the Computer: To keep things interesting, one of the players in the game in our version of the game will be the computer! Your game will have 2 human players and the computer. Since a player's success in this game relies heavily on how lucky they are, there is no need to add any intelligence or strategy to the computer's gameplay. The computer's dice rolls and bets can be randomized.

Player Names: At the start of the game, please prompt the users for their names. Rather than printing out information such as "it is player 1's turn…" or "player 1 rolled..", your message should print out "it is Becky's roll…" or "Becky rolled…".

Tips for Implementing the Assignment:

- You will need to use Python's built-in random function. Check out this website for details.
- You may choose to use dictionaries to keep track of the information related to each of the players. Each player should have their own dictionary that stores the data related to that player, such as their name and scores.
- When you are programming, it is never a good idea to just sit down and start coding. It is very important to plan out your solution. All of your code **MUST** be encapsulated within functions. When you are deciding what functions you will need to write, think about each of the individual tasks that your program will need to complete in order to implement the game in its entirety. Here are some suggestions to help get you started. Some of the tasks that your program will need to be able to do include…
    - roll a dice (in the case of this assignment, select a random number between 1 and 6).
    - re-roll the dice (if required, due to a fumble).
    - determine whether a player rolled a winning combination.
    - ask users to input their bets.

Important Details

All of your code **MUST** be encapsulated within functions. A "main" function will start the program execution. Solutions that are not written with functions will <u>not</u> receive a passing grade.

Your program must be well documented. Marks are allocated for documentation, neatly organized code, suitable variable names etc. Please refer to the program styling guide to see what we expect for proper documentation.

What to Submit

Your program should be put into one single Python file. Upload this file to OnQ.

Failure to submit the proper file will result in zero marks. Please be sure to check that you have submitted the correct version of your program.

Marking

The assignment will be marked using a rubric.

I do not give extensions on assignments. A 24 hour extension is available to everyone but comes with a penalty (see the rubric). Please do not ask for another extension unless you have a very, very good reason.

Academic Integrity

Please refer to the notes on Academic Integrity. It is ok (and encouraged) to talk about solutions with your friends but please do your own work. It is also ok to take snippits of code from the internet as long as the code does not solve the entire problem (which would be pretty hard in the case). If you use snippits of code, please document it in your comments. A breach of Academic Integrity will result in an F in this course. Please do your own work.

Due Date

Jul 16, 2021 11:30 PM

▼ Hide Rubrics

Rubric Name: assignment1Rubric

assignment1Rubric
— / 17

Submit Assignment

Files to submit *

(0) file(s) to submit

After uploading, you must click Submit to complete the submission.

Add a File　　　　**Record Audio**　　　　**Record Video**

Comments

	Paragraph			Font Famil	Font Size