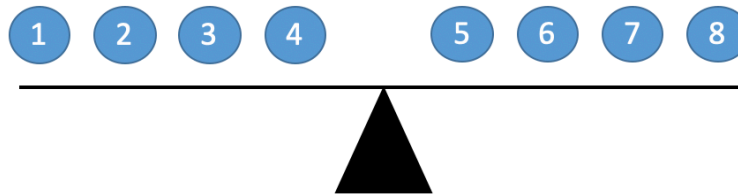# Odd Ball – 2022

OVERVIEW

In this assignment, you are going to design and develop a python program to run an interactive odd-ball game. The game is composed of an even number of balls and a single scaling device. All balls look identical in size and appearance, and all balls weigh exactly the same except one ball, which is heavier, called the odd ball. Balls are identified by their number labels in consecutive numerical sequence, starting with 1. The scale is a balance instrument with a left pan and a right pan used for weighing, the beam is in balance when the 2 pans contain exactly the same mass. The implemented program will interact with a player and the player will try to guess the odd ball by repeatedly weighing different balls on the scale. During the game the program and the player will take turns in an alternative manner: the player chooses some balls to weigh, the program outputs the result and then prompts the user to make a guess for the odd ball.

At the start of the program, it first prompts the player to enter the number of balls for the game. The number of balls must be even, minimum 2. After the prompt, the game randomly chooses one ball as the odd ball and keeps it secret from the player. Subsequently the game prompts the player for the balls to be placed on the left pan of the scale, and for the balls on the right. In return the game outputs the result of the weighing as whether the mass on both pans are balanced, left pan is heavier or right pan heavier. Finally, the player makes a guess for the odd ball. If the guess is not correct, the game will prompt the player to weigh again and the same process repeats until a correct answer is entered. When a correct guess is received, the game will output the total number of guesses made and total number of times the scale was used.

SCOPE

- At the start of the program, display a brief introduction about the game.
- For each game, prompt player for the number of balls, the number must be even, minimum 2.
- Implement the Game Cycle:
    a. Prompt user for ball weighing.
    b. Output the result of the scale
        i. Balanced
        ii. Left Heavier
        iii. Right Heavier
    c. Prompt user to make a guess for the odd number
- Track total number of times the scale used for each game.
- Display the scale usage count after the correct guess is entered.
- At the end of the game, prompt player to quit or start a new game
- Input Validation (User Interface)
    a. Ensure correct input is entered, if not, display prompt again after showing the warning.
    b. Robustness: program should not crash due to invalid input
- Coding Styles
    a. Ensure that your program follows the proper layout structure as discussed in class.
    b. You might declare global variables used for this assignment, ensure that a consistent naming convention is in place to differentiate various variable scopes.

NOTE:

- Keep your entire source code in ONE SINGLE file.
- Use only standard python modules
- In your design stick ONLY to functions, in other words, no class objects of your own.