Design of Lab Final – Jacob Bollinger

Problem:

Design software to play a dice game.

Task 1:

Define a function named robotTurn(). It will receive the inputs turnTotal and score. Set a variable lose to False. Create a while loop to run while turnTotal is less than twenty and lose is False. Inside set turnTotal equal to the function roll() with the input turnTotal. Outside of the while loop add turnTotal to score and set turnTotal to zero. Return turnTotal and score.

Define a function named roll(). It will take the input turnTotal. This function will set die equal to a random integer between 1 and 6 using randint. Print ‘Roll:’ and die. Use an if statement to test if die is greater than one. If it is, add die to turnTotal. Else set turnTotal to zero and lose to True. Return turnTotal.

Task 2:

Define a function named robotTurn(). It will receive the inputs turnTotal and score. Set a variable lose to False. Create a while loop to run while turnTotal is less than twenty and lose is False. Inside set turnTotal equal to the function roll() with the input turnTotal. Outside of the while loop add turnTotal to score and set turnTotal to zero. Return score.

Define a function named roll(). It will take the input turnTotal. This function will set die equal to a random integer between 1 and 6 using randint. Print ‘Roll:’ and die. Use an if statement to test if die is greater than one. If it is, add die to turnTotal. Else set turnTotal to zero and lose to True. Return turnTotal.

Define a function named main. Start it by initializing the integers robotScore and goalScore. Set a while loop to run while robotScore is less than goalScore. Inside set robotScore equal to robotTurn(robotTurnTotal, robotScore).

Task 3:

Define a function named displayScores() with the parameters robot1Score, and robot2Score. This function will print 1 line of ‘PC Score: ’, robot1Score, ‘ Mac Score: ’, robot2Score.

Define a function named robotTurn(). It will receive the inputs turnTotal and score. Set a variable lose to False. Create a while loop to run while turnTotal is less than twenty and lose is False. Inside set turnTotal equal to the function roll() with the input turnTotal. Outside of the while loop print ‘Turn total: ’ and turnTotal. Then add turnTotal to score and set turnTotal to zero. Return score.

Define a function named roll(). It will take the input turnTotal. This function will set die equal to a random integer between 1 and 6 using randint. Print ‘Roll:’ and die. Use an if statement to test if die is greater than one. If it is, add die to turnTotal. Else set turnTotal to zero and lose to True. Return turnTotal.

Define a final function named main. Start it by initializing the integers robot1Score, robot2Score, and goalScore. Set a while loop to run while robot1Score is less than goalScore and robot2Score is less than goalScore. Inside call displayScores() with inputs robot1Score and robot2Score. Print ‘It is PC’s turn.’. Set robot1Score equal to robotTurn(robot1TurnTotal, robot1Score). Print ‘It is Mac’s turn.’. Set robot2TurnTotal, robot2Score equal to robotTurn(robot2TurnTotal, robot2Score). Outside of the while loop use an if statement to test if robot1Score is greater than or equal to goalScore. If it is, print ‘PC wins.’. Else print ‘Mac wins.’.

Task 4:

Define a function named displayScores() with the parameters humanScore, and robotScore. This function will print 1 line of ‘Human Score: ’, humanScore, ‘ Mac Score: ’, robotScore.

Define a function named robotTurn(). It will receive the inputs turnTotal and score. Set a variable lose to False. Create a while loop to run while turnTotal is less than twenty and lose is False. Inside set turnTotal equal to the function roll() with the input turnTotal. Outside of the while loop print ‘Turn total: ’ and turnTotal. Then add turnTotal to score and set turnTotal to zero. Return score.

Define a function named humanTurn(). It will receive the inputs turnTotal and score. Set a variable lose to False. Create a while loop to run while done is False and lose is False. Inside set a variable named move equal to input(‘Turn Total: ’, turnTotal, ‘ Roll/Hold?’). Create an if statement to test if move is equal to ‘’. If it is, set turnTotal equal to the function roll() with the input turnTotal. Else set done equal to True. Outside of the while loop print ‘Turn total: ’ and turnTotal. Then add turnTotal to score and set turnTotal to zero. Return score.

Define a function named roll(). It will take the input turnTotal. This function will set die equal to a random integer between 1 and 6 using randint. Print ‘Roll:’ and die. Use an if statement to test if die is greater than one. If it is, add die to turnTotal. Else set turnTotal to zero and lose to True. Return turnTotal.

Task 5:

In the top left corner of the window will be the scores of both players. When the human is playing, there will be a turn total in the bottom left of the window. There will be 2 buttons in the bottom center of the window side by side with the left one reading ‘roll’ and the right one reading ‘hold’. In the top right will be a log of what the robot did in its last turn.