Python Programming #3B

The TLCW Nim Game

Writing a computer program to play the TLCW nim game like an ordinary person

Overview: In this programming assignment, you need to write a program that can interact with the user to define the initial setting of a simple nim game and can then play the game following the rules. Read the description of the TLCW nim game below to understand the rule of the simple nim game first, and then develop a computer program that plays the TLCW nim game just like an ordinary player.

The TLCW nim game (taking the last coin to win the game):

- Two players take turn to play the game with a pile of coins.
- Before the game starts, the players negotiate to determine (i) n, the number of coins to begin with, and (ii) m, the maximum number of coins a player can take on each turn where *m is less than n and is at least 1*.
- The players then take turn to take at least one coin but no more than *m* coins each time until there is no coin left. Whoever takes the last coin is the winner of the game.

Steps for developing a computer program that plays the TLCW nim game like an ordinary player:

1. In the beginning, your program should ask the user to enter the information of two natural numbers ***n*** and ***m*** where ***n*** *is* the number of coins in the beginning while ***m*** is the maximum number of coins a

player can take on each turn. Your program should read and store the numerical values of ***n*** and ***m*** accordingly into two variables ***numberOfCoinsNow*** and ***maximum*** in the program.

2. Your program should then ask the user to enter the information of either **C** or **U** to indicate whether user wants the computer (**C**) or the user (**U**) to start the game. Your program should then read and store the information from the user and store it accordingly into a variable ***currentTurn*** in the program.
3. Now in your program write the following statement `import random` to import the support of random number generation so that we can generate random moves in the following of the program.
4. Write a *while* loop to repeatedly do the following things in each iteration as long as the number of coins (i.e. the value of the variable ***numberOfCoinsNow***) is greater than 0:
   - i. **Report the status**: Print a message to report **the number of coins left** at the moment (i.e. the value of the variable ***numberOfCoinsNow***), and **the player to make the move** this time according to. the contents of the variable ***currentTurn*.**
   - ii. **If it is the computer's turn:** If the value stored in the variable ***currentTurn*** indicates it is the computer's turn now (i.e. the contents of the variable ***currentTurn*** is the string "C"):
     - a) check whether the number of coins left coins (i.e. the value of the variable ***numberOfCoinsNow***) is **no more than** the maximum number of coins allowed to take on each turn (i.e. the value of the variable ***maximum***). If this is true, print out a message to inform the user that the computer will make a move to take away all the coins left

to win the game. And then update the value of ***numberOfCoinsNow*** to 0 as a result of that move.

b) check whether the number of coins left coins (i.e. the value of the variable ***numberOfCoinsNow***) is **more than** the maximum number of coins allowed to take on each turn (i.e. the value of the variable ***maximum***). If this is true, write the following statement `move = random.randint(1,` ***maximum*** `)` to generate a random number in the range of 1 to ***maximum*** and store the result in the variable ***move***. Print out a message to inform the user that the computer will make a move to take away that many coins away. And then then update the value of ***numberOfCoinsNow*** to ***numberOfCoinsNow - move*** as a result of that move.

iii. **If it is the user's turn:** If the value stored in the variable ***currentTurn*** indicates it is the user's turn now (i.e. the contents of the variable ***currentTurn*** is the string "U"):

a) Ask the user to enter the number of coins he/she would like to take away. Read the user input and convert it into a numerical value to be stored in the variable `move`.

b) Check whether the proposed move is a valid move, in other words, whether ***move*** is no more than number of coins left coins (i.e. the value of the variable ***numberOfCoinsNow***) and is in the range of 1 to ***maximum*** also. If it is a valid move, print out a message to acknowledge that it is a move to go and then update the value of ***numberOfCoinsNow*** to

numberOfCoinsNow - `move` as a result of that move. If it is a **not** valid move, update the value of ***numberOfCoinsNow*** to 0 to end the game abruptly and print out a message to inform the user that he/she is cheating and has lost the game because of it.

iv. **Switch the turn**: If it is the computer's turn now (i.e. the contents of the variable ***currentTurn*** is the string "C"), switch it to the user's turn by write the following statement ***currentTurn*** **=** "U". Otherwise, it is the user's turn now (i.e. the contents of the variable ***currentTurn*** is the string "U") and we should switch it to the user's turn by write the following statement ***currentTurn*** **=** "C".

5. After the loop, use an *if else* statement to determine the winner of the game and print out a message to congratulate the winner and thank the user for playing the game.
   i. Check whether it is the computer's turn now.
   ii. If so, then the user is the last player who took away the last coin, and thus the computer is the winner. Print out a congratulation message to the user.
   iii. Otherwise the computer is the last player who took away the last coin, and thus the user is the winner. Print out a message to thank the user for playing the game.