CompSci 101 - Assignment 02 *Ventuno*

Due: 4:30pm, 30th April 2020.

Worth: This assignment is marked out of 30 and is worth 3% of your final mark.

Topics covered:

- Functions, if statements, string manipulation, converting types

The work done on this assignment **must** be your own work. Think carefully about any problems you come across, and try to solve them yourself before you ask anyone else for help. Under no circumstances should you use code written by another person in your assignment solution or give your code to another student.

VERY IMPORTANT:

This assignment has three sections.

Section A is marked using Coderunner3 (20 marks). For this section you need to define 14 functions. The 14 functions, when inserted into the final skeleton program, create the "one player against the computer" game of Ventuno (see the game description below). Each function is described in Section A of this document and there are three Python skeleton programs (`Stage01_Ventuno.py, Stage02_Ventuno.py, Stage03_Ventuno.py`) which you **MUST** use to develop the 14 functions.

Section B (2 marks).
Assign **your** username to the `username` variable in the `main()` function of the Assignment 2 program. Insert the 14 functions from Section A into the Assignment two program. The program should execute without error allowing the user to play against the computer.

Section C (5 marks).
Section C of this assignment creates a more interesting game of `Ventuno` where the computer player makes more sensible decisions. The `Stage04.py` skeleton program must be used to develop function 15.

Submission

Section A - Functions 1 - 14 are submitted using CodeRunner3.

Section B and Section C - Submit your completed Assignment 2 Python program (just one Python file named `"YourUsernameA2.py"`, e.g., `afer023A2.py`) using the Assignment Dropbox:

https://adb.auckland.ac.nz/Home/

NOTES:

- This assignment is marked out of 30 and is worth 3% of your final mark. Three marks out of 30 are assigned for the style of your program (program docstring, good variable names, etc.).
- You must only use the features taught in CompSci 101.
- An example output using the completed program is available at the end of this document:

https://www.cs.auckland.ac.nz/courses/compsci101s1c/assignments/

Assignment 2 – The game of Ventuno

In this assignment, you will develop the `Ventuno` game which is a dice throwing game played against the computer. The object of the game of `Ventuno` is to reach a score (by throwing a dice) which is higher than the other player's score, but without going over 21. If you go over 21, you "bust" and automatically lose the game. When having a turn, the player can choose to either roll the dice and add that score to their current score or the player can choose to "stay" at their current score and in this case their score will not change for the rest of the game.

Assignment 2 Section A (20 marks)

Download the Python skeleton programs (`Stage01.py, Stage02.py, Stage03.py`) from the CompSci 101 assignments website:

`https://www.cs.auckland.ac.nz/courses/compsci101s1c/assignments/`

These programs will be used to develop the 14 functions used in the Ventuno game.

- Use the `Stage01.py` program to complete and test functions 1, 2, 3 and 4.
- Use the `Stage02.py` program to complete and test functions 5, 6, 7, 8 and 9.
- Use the `Stage03.py` program to complete and test functions 10, 11, 12, 13 and 14.

Once you are happy that a function executes correctly, submit the whole function to CodeRunner3:

https://coderunner3.auckland.ac.nz/moodle/

Submission

When you press the `Check` button in CodeRunner3, you will receive immediate feedback telling you if you have passed all the tests for that function. You can submit as many times as you need. You need to submit each function separately. Your marks for Section A of Assignment 2 are obtained through CodeRunner3. When you have successfully completed all the functions and your code passes all the tests click the `"Submit all and finish"` button.

Once you have successfully completed Sections A, B and C (or Sections A and B) for this assignment you should submit your final program in a module (file), named `"YourUsernameA2.py"`, e.g. `afer023A2.py`.

Submit your program using the Assignment Dropbox: https://adb.auckland.ac.nz/Home/

Section A, Stage 1 (Use the `Stage01.py` program to develop these four functions)

1.

Define the `display_line_of_symbols(indent, symbol, how_many)` function. This function is passed three string parameters: a string of spaces (`indent`), a single character (`symbol`) and an integer (`how_many`). The function prints the `indent` parameter followed by a row of symbols all in one line of output. The number of symbols is given by the `how_many` parameter.

Note: the `indent` parameter could be the empty string ("").

<-------------------------->

2.

Define the `display_introduction(indent, username, name)` function. This function is passed three strings as parameters, a string of blank spaces, a string representing a username (e.g. `'afer023'`) and the name of the player. This function prints six lines of output:

- Lines 1 and 6. are blank lines,
- Lines 2 and 5. are rows of 27 `'='` symbols preceded by the `indent` parameter. Your code **MUST** make two calls to the `display_line_of_symbols()` function developed in Question 1 to do this.
- Line 3 prints the `indent` parameter followed by the string `'Ventuno written by '` followed by the `username` parameter.
- Line 4 prints the `indent` parameter followed by the string `'Welcome  '` followed by the `name` parameter.

Note: you can assume that the `display_line_of_symbols()` function is available and you should not include it in your answer when you test this function in CodeRunner.

<-------------------------->

3.

Define the `get_play_menu_selection(indent)` function. This function is passed a string of spaces (`indent`) as a parameter. The function prints four lines of output and returns the integer entered by the user. **Each** line printed is preceded by the `indent` parameter.

- Line 1 is a blank line,
- Line 2 prints the string `'1. ROLL'`,
- Line 3 prints the string `'2. STAY'`,
- Line 4 displays 3 spaces followed by the prompt `'Enter selection: '` and the function returns the integer value entered by the user.

<-------------------------->

4.

Define the `get_main_menu_selection(indent)` function. This function is passed a string of spaces (`indent`) as a parameter. The function prints five lines of output and returns the integer entered by the user. **Each** line printed is preceded by the `indent` parameter.

- Line 1 is a blank line,
- Line 2 prints the string `'1. PLAY A NEW GAME'`,
- Line 3 prints the string `'2. SEE STATS'`,
- Line 4 prints the string `'0. QUIT'`,
- Line 5 displays 3 spaces followed by the prompt `'Enter selection: '` and the function returns the integer value entered by the user.

<-------------------------->

Section A, Stage 2 (Use the `Stage02.py` program to develop these five functions)

5.

Define the `display_current_player(current_player)` function. This function is passed a string parameter, the player's name (`current_player`). The function prints the `current_player` parameter string followed by the string `"'s turn:"`.

<-------------------------->

6.

Define the `get_next_player(current_player, player_name)` function. This function is passed two string parameters: the current player's name and the name of the non-computer player. If the `current_player` parameter is the same as the string, `"Computer"`, the function returns the `player_name` parameter, otherwise, the function returns the string, `"Computer"`.

<-------------------------->

7.

Define the `get_random_starting_player_name(player_name)` function. This function is passed the name of the non-computer player as a parameter. The function randomly returns either the `player_name` parameter or the string, `"Computer"`.

You can assume that the `random` module has been imported.

IMPORTANT: In order to pass the CodeRunner3 tests, you need to generate a random number which is either a 1 or a 2 using the `random.randrange()` function and you should return the `player_name` parameter if the random number is a 1, otherwise return the string, `"Computer"`.

<-------------------------->

8.

Define the `add_dice_roll(current_score, current_player)` function. This function is passed two parameters: an integer and the name of the current player. The function generates a random dice value between 1 and 6 (both inclusive) using the `random.randrange()` function, prints the `current_player` parameter, followed by `"'s dice roll: "` followed by the generated dice value.

Finally the function returns an integer: the sum of the `current_score` parameter and the dice value.

Note: You can assume that the `random` module has been imported.

<-------------------------->

9.

Define the `get_starting_score()` function. This function returns a random number which is either 12, 13, 14, 15 or 16, i.e. at each call, this function is equally likely to return either 12, 13, 14, 15 or 16.

Notes
You can assume that the `random` module has been imported,
You **MUST** use the `random.randrange()` function to generate the random number.

<-------------------------->

Section A, Stage 3 (Use the `Stage03.py` program to develop these five functions).

IMPORTANT: Firstly, copy the `display_line_of_symbols()` function from Question 1. and paste it into the `Stage03.py` file.

10.

Define the `display_current_scores(indent, score_player, score_computer, player_stays, computer_stays, player_name)` function.

This function is passed six parameters: a string of spaces (`indent`), the player's score (`score_player`), the computer's score (`score_computer`), two boolean parameters indicating whether the player has 'stayed' and whether the computer has 'stayed' (`player_stays`, `computer_stays`) and a string (`player_name`). The function prints eight lines of output:

- Lines 1 and 8 are blank lines,
- Lines 2, 3, 6 and 7 are rows of 37 `'-'` symbols preceded by the `indent` string. Your code **MUST** make four calls to the `display_line_of_symbols()` function developed in Question 1.
- Line 4 prints the `indent` parameter followed by the player's name, followed by the string `"'s score: "`, followed by the player's current score. If the player has stayed, this string is followed by `' - '` followed by the player's name and the string `'stays'`.
- Line 5 prints the `indent` parameter, followed by the string `'Computer's score: '`, followed by the computer's current score. If the computer has stayed, this string is followed by `' - Computer stays'`.

<-------------------------->

11.

Define the `game_is_over(score_player, score_computer, player_stays, computer_stays)` function. This function is passed four parameters: the player's score (`score_player`), the computer's score (`score_computer`), the computer's score (`score_computer`), two booleans indicating whether the player has 'stayed' and whether the computer has 'stayed' (`player_stays`, `computer_stays`). This function returns `True` if the game is over, `False` otherwise. The game is over if both players have stayed or if either of the players has a score which is greater than 21.

<-------------------------->

12.

Define the `get_winner_name(score_player, score_computer, player_name)` function. This function is passed three parameters: the player's score (`score_player`), the computer's score (`score_computer`), the player's name (`player_name`). The function returns the name of the winner. If the two scores are equal, the result is a draw and the function returns the string, `"draw"`. If the player's score is over 21 the computer wins and the function returns the string `"Computer"`. If the computer score is over 21 the player wins and the function returns the player's name. In all other cases the player (or `"Computer"`) with the highest score wins.

<-------------------------->

13.

Define the `display_game_result(indent, score_player, score_computer, game_winner, player_name)` function. This function is passed five parameters: a string of spaces (`indent`), the player's score (`score_player`), the computer's score (`score_computer`), the name of the winner (`game_winner`) and the player's name (`player_name`). The function prints eight lines of output:

- Line 1 is a blank line,
- Lines 2, 3, 7. and 8 are rows of 25 `'+'` symbols preceded by the `indent` string. Your code **MUST** make four calls to the `display_line_of_symbols()` function developed in Question 1.
- Line 4 prints the `indent` parameter followed by the player's name, followed by the string `"'s score: "`, followed by the player's score.
- Line 5 prints the `indent` parameter, followed by the string `'Computer's score: '`, followed by the computer's score.
- Line 6 prints the `indent` parameter, followed by one of the following strings:
  - `'Computer has won.'` if the `game_winner` parameter is the string, `"Computer"`,
  - the player's name followed by the string `' has won.  Well done!'` if the `game_winner` parameter is the same as the `player_name` parameter,
  - `'Result is a draw.'` if the `game_winner` parameter is the string, `"draw"`.

<-------------------------->

14.

Define the `display_statistics(player_name, total_number_of_games, games_won_by_player, games_won_by_computer)` function. This function is passed four parameters: the player's name (`player_name`), the total number of games played (`total_number_of_games`), the number of games won by the player (`games_won_by_player`), the number of games won by the computer (`games_won_by_computer`) and the player's name (`player_name`). The function prints eleven lines of output:

- Lines 1 and 8 are blank lines,
- Lines 2, 3, 10. and 11. are rows of 32 `'*'` symbols.
- Line 4 prints the `"   Number of games played: "` string, followed by the total number of games played.
- Line 5 prints the string `"   Games won by "` followed by the player's name, followed by `": "` followed by the number of games won by the player.
- Line 6 prints the string `"   Games won by Computer: "` followed by the number of games won by the computer.
- Line 7. prints the string `"   Games resulting in a draw: "` followed by the number of games which were a draw.
- Line 9. prints whichever one of the following strings is applicable:
  - `"***  "` followed by the player's name, followed by the string `" is winning by "` followed by the number of games by which the player is leading and finally the string, `" *** "`.
  - `"***  Computer is winning by "` followed by the number of games by which the computer is leading and finally the string `" ***"`.
  - `"*** Final result is a draw ***"`.

<-------------------------->

Assignment 2 Section B (2 marks)

Download the Python skeleton program (`SkeletonA2.py`) from the CompSci 101 assignments website:

```
https://www.cs.auckland.ac.nz/courses/compsci101s1c/assignments/
```

and rename the `Ventuno.py` file to `"YourUsernameA2.py"`, e.g., `afer023A2.py`.

Change the **first** statement in the `main()` function so that **your** own username is assigned to the variable, `username`:

```
username = "afer023" #change this string to your own username
```

Insert the completed 14 functions from Section A into this program and run the program. Your Assignment 2 program should execute without error. Now you will be able to play a very basic version of the game of `Ventuno` against the computer. You are welcome to change the name of the player. In this version of the game, the user is playing against the computer and the computer player **always** 'chooses' to stay.

Some parts of the `main()` and the `play_a_game()` functions are shown on Pages 8 and 9 of this document. These two functions have already been defined for you and should not be changed in any way except for the change indicated in Section B.

<-------------------------->

Assignment 2 Section C (5 marks)

Section C, Stage 4 (Use the `Stage04.py` program to develop this function)

In the Assignment 2 program, the `get_computer_selection()` function is very basic. This is the function which is called when it is the computer's turn to play and the computer has to decide whether to roll the dice or to 'stay'. Currently the function always returns the number 2, indicating that the computer chooses to 'Stay' whenever it is the computer's turn. This makes the game very tedious and predictable.

Download the `Stage04.py` skeleton program from the CompSci 101 assignments website:

```
https://www.cs.auckland.ac.nz/courses/compsci101s1c/assignments/
```

and use this program to develop the `get_computer_selection()` function used in the `Ventuno` game. When you are happy that this function executes correctly, copy the whole function into your Python program, test the program and submit it to the dropbox.

15.

Complete the `get_computer_selection(score_player, score_computer, player_stays)` function so that the computer player makes sensible decisions when it is the computer player's turn to play.

This function is passed three parameters: the player's score (`score_player`), the computer's score (`score_computer`) and a `boolean` parameter indicating whether the player has stayed or not. Based on this information the function should decide whether the computer should roll the dice or stay. The function returns the number 1 if the computer should roll the dice and the function returns the number 2 if the computer should stay.

You may use any algorithm you like to decide the computer move but the algorithm must be sensible. You may wish to add some randomness to the computer's decision, e.g. if both players have a score of 18, should the computer roll or stay? Some considerations:

If the player score is higher (but equal to 21 or less) than the computer score, then it is sensible for the computer to choose to roll the dice.

If the player has stayed and the player score is lower than the computer score, then it is sensible for the computer to choose to stay.

If the computer score is the same as the player score and both scores are close to 21 then it may be sensible for the computer to choose to stay because it is probable that another dice roll will take them above 21. Your algorithm may use some randomness to decide in this case.

My algorithm makes the choices shown in response to the following four function calls:

```
selection1 = get_computer_selection(19, 18, True) #ROLL
selection2 = get_computer_selection(20, 20, True) #STAY
selection3 = get_computer_selection(18, 18, True) # STAY
selection4 = get_computer_selection(18, 18, False) #ROLL
```

<-------------------------->

Once you have completed function 15, copy this function into the YourUsernameA2.py program, test your program and submit it to the dropbox.

Some parts of the skeleton program:

```
import random

def main():
    username = "afer023" #change this to your username
    player_name = "Adriana" #change this name
    . . .
   while selection !=  option_quit:
        selection = get_main_menu_selection(" " * 5)
        if selection == option_new_game:
            number_of_games += 1
            winner_name = play_a_game(player_name)
            if winner_name == player_name:
                won_by_user += 1
            elif winner_name == "Computer":
                won_by_computer += 1
        elif selection == option_see_stats:
            display_statistics(player_name, number_of_games, won_by_user,
                                                                 won_by_computer)
            . . .
```

```
def play_a_game(player_name):
    ventuno = 21
    player_stays = False
    computer_stays = False
    roll = 1
    stay = 2
    score_player = get_starting_score()
    score_computer = get_starting_score()
    . . .
    while not game_is_over(score_player, score_computer, player_stays,
                                                         computer_stays):
        if current_player == "Computer":
            if not computer_stays:
                display_current_player(" " + current_player)
                selection = get_computer_selection(score_player, score_computer,
                                                               player_stays)
                if selection == roll:
                    score_computer = add_dice_roll(score_computer, "  Computer")
                    if score_computer == ventuno:
                        computer_stays = True
                else:
                    computer_stays = True
        elif not player_stays:
            display_current_player(" " + current_player)
            selection = get_play_menu_selection(' ' * 3)
            print()
            if selection == roll:
                score_player = add_dice_roll(score_player, " " + player_name)
                if score_player == ventuno or (computer_stays and score_player >
                                                         score_computer):
                    player_stays = True
            elif selection == stay:
                player_stays = True
        display_current_scores(" " * 5, score_player, score_computer,
                               player_stays, computer_stays, player_name)
        current_player = get_next_player(current_player, player_name)

    game_winner = get_winner_name(score_player, score_computer, player_name)
    display_game_result("  ", score_player, score_computer, game_winner,
                                                               player_name)
    return game_winner
```

Example Output

Below are some sections of output produced by the completed program. Your program must execute in the way described and the output should have the same format as the output below. The player input is shown in a larger bold font.

```
     ===========================
     Ventuno written by afer023
     Welcome Adriana
     ===========================


       1. PLAY A NEW GAME
       2. SEE STATS
       0. QUIT
          Enter selection: 1

       ---------------------
       ---------------------
       Adriana's score: 12
       Computer's score: 13
       ---------------------
       ---------------------

   Adriana's turn:

     1. ROLL
     2. STAY
        Enter selection: 1

   Adriana's dice roll: 6

       ---------------------
       ---------------------
       Adriana's score: 18
       Computer's score: 13
       ---------------------
       ---------------------

   Computer's turn:
    Computer's dice roll: 4

       ---------------------
       ---------------------
       Adriana's score: 18
       Computer's score: 17
       ---------------------
       ---------------------

   Adriana's turn:
      1. ROLL
      2. STAY
         Enter selection: 2


        -------------------------------------
        -------------------------------------
        Adriana's score: 18 - Adriana stays
        Computer's score: 17
        -------------------------------------
        -------------------------------------

    Computer's turn:
     Computer's dice roll: 3

        -------------------------------------
        -------------------------------------
        Adriana's score: 18 - Adriana stays
        Computer's score: 20
        -------------------------------------
        -------------------------------------



        -------------------------------------
        -------------------------------------
        Adriana's score: 18 - Adriana stays
        Computer's score: 20
        -------------------------------------
        -------------------------------------

    Computer's turn:

        -------------------------------------
        -------------------------------------
        Adriana's score: 18 - Adriana stays
        Computer score: 20 - Computer stays
        -------------------------------------
        -------------------------------------


     +++++++++++++++++++++++++
     +++++++++++++++++++++++++
     Adriana's score: 18
     Computer's score: 20
     Computer has won.
     +++++++++++++++++++++++++
     +++++++++++++++++++++++++

        1. PLAY A NEW GAME
        2. SEE STATS
        0. QUIT
        Enter selection: 1

      ---------------------
      ---------------------
      Adriana's score: 14
      Computer's score: 14
      ---------------------
      ---------------------

Adriana's turn:

  1. ROLL
  2. STAY
     Enter selection: 1

Adriana's dice roll: 6

      ---------------------
      ---------------------
      Adriana's score: 20
      Computer's score: 14
      ---------------------
      ---------------------

Computer's turn:
 Computer's dice roll: 5

      ---------------------
      ---------------------
      Adriana's score: 20
      Computer's score: 19
      ---------------------
      ---------------------

Adriana's turn:

  1. ROLL
  2. STAY
     Enter selection: 2


      -------------------------------------
      -------------------------------------
      Adriana's score: 20 - Adriana stays
      Computer's score: 19
      -------------------------------------
      -------------------------------------

Computer's turn:
 Computer's dice roll: 6
```

```
      -------------------------------------
      -------------------------------------
      Adriana's score: 20 - Adriana stays
      Computer's score: 25
      -------------------------------------
      -------------------------------------


   +++++++++++++++++++++++++
   +++++++++++++++++++++++++
   Adriana's score: 20
   Computer's score: 25
   Adriana has won.  Well done!
   +++++++++++++++++++++++++
   +++++++++++++++++++++++++

      1. PLAY A NEW GAME
      2. SEE STATS
      0. QUIT
         Enter selection: 1
```

Play two more games – the output is not shown here

```
      1. PLAY A NEW GAME
      2. SEE STATS
      0. QUIT
         Enter selection: 2

 ********************************
 ********************************
   Number of games played: 4
   Games won by Adriana: 3
   Games won by Computer: 1
   Games resulting in a draw: 0

 *** Adriana is winning by 2 ***
 ********************************
 ********************************

      1. PLAY A NEW GAME
      2. SEE STATS
      0. QUIT
         Enter selection: 1
```

Play another game – the output is not shown here

```
      1. PLAY A NEW GAME
      2. SEE STATS
      0. QUIT
         Enter selection: 0
```

```
*******************************
*******************************
  Number of games played: 5
  Games won by Adriana: 4
  Games won by Computer: 1
  Games resulting in a draw: 0

*** Adriana is winning by 3 ***
*******************************
*******************************

Thank you for playing Ventuno Adriana
```