Assignment 3: due 8am on Mon, Oct 29, 2018

Summary of Instructions

Note	Read the instructions carefully and follow them exactly
Assignment Weight	6% of your course grade
Due Date and time	8am on Monday, Oct 29, 2018
Important	As outlined in the syllabus, late submissions will not be accepted.
	Any files with syntax errors will automatically be excluded from grading. Be sure to test your code before you submit it
	For all functions, both in Part 1 and 2, make sure you've written good docstrings that include type contract, function description and the preconditions if any.

This is an individual assignment. Please review the Plagiarism and Academic Integrity policy presented in the first class, i.e. read in detail pages 15 – 18 of course outline (ITI1120-course-outline-syllabus-Fall2018.pdf). You can find that file on Brightspace under Course Info. While at it, also review Course Policies on pages 13 and 14.

The goal of this assignment is to learn and practice the concepts covered thus far: function design, lists and loops.

What needs to be submitted is explained next.

The assignment has two parts. Each part explains what needs to be submitted. Put all those required documents into a folder called a3_xxxxxx where you changed xxxxxx to your student number, zip that folder and submit it as explained in Lab 1. In particular, the folder should have the following files:

In part 1 you will implement 3 small programs. In part 2, you will implement a card game. Put all the below four required documents into a folder called HW3_xxxxxx.zip that folder and submit it by mailing it to drophw@gmail.com. Put in the SUBJECT: HW3-yourLabSection-YourStudentNumber. (Ex: HW3-C3-1234567). (In particular, the folder (and thus your submission) should have the following files:
Part 1: HW3_Q1_xxxxxx.py, HW3_Q2_xxxxxx.py, HW3_Q3_xxxxxx.py
Part 2: HW3_GAME_xxxxxx.py

Both of your programs must run without syntax errors. In particular, when grading your assignment, TAs will first open your file, say a3_GAME_xxxxxx.py with IDLE and press Run Module. If pressing Run Module causes any syntax error, the grade for Part 2 becomes zero. The same applies to Part 1.

Furthermore, for each of the functions (in Part 1 and Part 2), I have provided one or more tests to test your functions with. To obtain a partial mark your function may not necessarily give the correct answer on these tests. But if your function gives any kind of python error when run on the tests provided below, that question will be marked with zero points. Test/example runs for each function in part 2 are given inside of its docstrings in the provided starter file HW3_game_xxxxxx.py.

To determine your grade, your functions will be tested both with examples provided in Part 1 and HW3_game_xxxxxx.py and with some other examples. Thus you too should test your functions with more example than what I provided in Part 1 and HW3_game_xxxxxx.py.

Global variables are not allowed. If you do not know what that means, for now, interpret this to mean that inside of your functions you can only use variables that are created in that function. For example, this is not allowed, since variable x is not a parameter of function a_times(a) nor is it a variable created in function a_times(a). It is a global variable created outside of all functions.

```
def a_times(a):
    result=x*a
    return result

x=float(input("Give me a number: "))
print(a_times(10))
```

1 Part 1 20 points

For this part of the assignment, you are required to write three short programs. For this part you need to submit 3 files: `a3_Q1_xxxxxx.py`, `a3_Q2_xxxxxx.py`, and `a3_Q3_xxxxxx.py`.

1.1 Question 1: (5 points)

Implement a Python function called `count_pos` that takes a list of numbers as an input parameter and returns the number of elements of that list that are positive (i.e. > 0). Then, in the `main`, your program should ask the user to input the list, then it should call `count_pos` function with that list, and print the result. In this question you may assume that the user will follow your instructions and enter a sequence of numbers separated by spaces. You can use str method `.strip` and `.split` to handle the user input. Here is a way to ask a user for a list:
`raw_input = input("Please input a list of numbers separated by space: ").strip().split()` But now `raw_input` is a list of strings that look like numbers so you need to create a new list that is a list of equivalent numbers.

Three examples of program runs:

```
Please input a list of numbers separated by space: 2 3.5 -1 -100
There are 2 positive numbers in your list.

Please input a list of numbers separated by space: 1 0 22 0 1
There are 3 positive numbers in your list.

Please input a list of numbers separated by space:
There are 0 positive numbers in your list.
```

Function call example:

```
>>> count_pos( [1, 0, 22.2, 0, 1.0, -10.5] )
3
```

1.2 Question 2: (5 points)

Implement a Python function called `two_length_run` that takes a list of numbers as input parameter and returns `True` if the given list has is at least one run (of length at least two), and `False` otherwise. Make sure the function is efficient (i.e. it stops as soon as the answer is known). Then, in the `main`, your program should ask the user to input the list, then it should call `two_length_run` function, and print the result. You can obtain a list of numbers from the user as explained in Question 1.
Four examples of program runs:

```
Please input a list of numbers separated by space: 1     4   3 3  4
True

Please input a list of numbers separated by space: 1 2 3      3   3 4.5 6 5
True

Please input a list of numbers separated by space: 1.0 2 3.7 4 3 2
False

Please input a list of numbers separated by space: 7.7
False
```

Function call example:

```
>>> two_length_run( [2.7, 1.0, 1.0, 0.5, 3.0, 1.0] )
True
```

1.3 Question 3: (10 points)

As mentioned, a run is a sequence of consecutive repeated values. Implement a Python function called `longest_run` that takes a list of numbers and returns the length of the longest run. For example in the sequence: `2, 7, 4, 4, 2, 5, 2, 5, 10, 12,` the longest run has length 4. Then, in the `main`, your program should ask the user to input the list, then it should call `longest_run function`, and print the result. You can obtain a list of numbers from the user as explained in Question 1.
Five examples of program runs:

```
Please input a list of numbers separated by space: 1 1 2 3.0 3 3 3 3 6 5
5

Please input a list of numbers separated by space:   6 6   7 1 1 1 1 4.5 1
4

Please input a list of numbers separated by space: 6 2.4 4 8 6
1

Please input a list of numbers separated by space: 3
1

Please input a list of numbers separated by space:
0
```

Function call example:

```
longest_run([6, 6, 7, 1.0, 1.0, 1.0, 1, 4.5, 1])
4
```

2 Part 2: Single Player Rummy Game with Dice and strange deck (80 points)

To clarify Part 2 specifications, I have provided sample tests for each required function inside of its docstrings in a3_game_xxxxxx.py. Furthermore, you can find example run of the whole game below and its associated video at the link below. The behaviour implied by the sample tests/runs and the video should be considered as required specifications in addition to what is explained in this document.

Here is the link to the video: https://youtu.be/Zw12qTyPnHo

Description:

A card in a standard deck has a suit (in particular, one of four suits: ♥, ♠, ♦, ♣) and a rank (one of 13 ranks: A, 2, 3, ..., 10, J, Q, K). Taking every pair of a suit and a rank gives rise to a standard deck of (4x13=) 52 cards. Imagine you have access to only an old fashion terminal that cannot display fancy characters like: ♥, ♠, ♦, ♣ but yet you would like to make a card game. You would first need to decide how to represent a card. One way to do that would be to represent a card by a 3 digit integer where the first digit (1 to 4) represents a suit and the two last digits (1 to 13) represent ranks. Let's call such a deck, a *strange* deck.

For part 2 of the assignment, you will need to make a (heavily) modified version of Rummy card game with this strange deck. In Rummy, the main goal is to build *melds* which consists of sets, two, three or four *of a kind* of the same rank; or *progression*, three or more cards in a sequence of consecutive ranks, of the same suit. So the set ♥10, ♦10, ♣10 forms three of a kind. And the set/sequence 7♦, 8♦, 9♦, 10♦, 11♦ forms a progression. In our strange deck, 210, 110, 310 would form three of a kind (since the first digit is a suit, so the ranks are 10, 10, 10) and the set 309, 307, 311, 308, 310 is a progression (since they all have a suit 3 and 07, 08, 09, 10, 11 is sequence of consecutive integers). Note that 201, 302, 303 is not a progression. Although 01, 02, 03 is a sequence of consecutive integers, the three cards do not have the same suit (some have suit 2 and some 3) so this is not a progression.

So the game that you will develop needs to go as follows: (The goal of the player is to get rid of all of her cards in as few rounds of the game as possible)

```
Step 0: The strange deck  is created and shuffled. (In order to test your game more
quickly you can reduce the numer of ranks to less than 13 and more  than 3). In your
implementation the created deck needs to be a list  of integers
representing a strange deck. Top of the deck is considered the last card in the  list.

Step 1:  The  player is dealt 7 cards  from the top of the strange deck.

Step 2: The following steps  are repeated until the player runs out of
the cards:

      Step 3: The player  rolls a dice
      If the player gets a 1:
          The player can discard any one card she likes. After that, the current round
          is over and the game goes back to Step 2.
```

```
      If instead the player gets num=2,3,4,5 or 6:
          The player if first delt, from the top of the deck,
          num or len(deck) cards, whichever is smaller.

          The player then keeps on discarding melds from her hand
          until she has no more melds. You program has to check
          that the set of cards that the player chooses indeed forms a
          valid meld before discarding them from the player's hand.
          Once she decides she is out of melds, the round is over
          and the game goes back to Step 2.

(Note that once the deck is empty and the player has no more melds,
no melds can ever be created again. Thus the player has to wait for 1 on the
dice. In order to avoid that unpleasantry your game should roll 1
i.e. set num to 1, in each round that starts with empty deck.)

Finally, once the player is out of cards, the total number of rounds
is reported.
```

As usual, whenever you ask the player for some input you should make sure they give you the required kind of input. You may assume that the player will follow instructions and give you a correct type of date but not the correct values. For example, if you are asking for an integer between 3 and 99, you may assume that the player will give you an integer but not that she will give you an integer in the correct range. Thus you should keep on repeating the question until you get a valid answer. Similarly if you ask the player for a meld, you may assume that the player will give you a set of 3 digit integers, but you will need to test if these cards are indeed in the players hand and that they form a meld.

As in the previous assignment, I provided you with the starter code in the file called, `HW3_game_xxxxxx.py`. Start by replacing `xxxxxx` in the file name with your student number. Then open the file. Your solution (code) for this part must go into that file in the clearly indicated spaces only. You are not allowed to delete or comment-out or change any parts of the provided code except for the keywords `pass`.

In `HW3_game_xxxxxx.py`, there are many functions with code missing. You must complete and use all of those functions. Your function should behave as explained in the function description and the example runs. All rea given in the docstrings. You also need to complete the `main`.

2.1 Testing Part 2

Here is what pressing Run on your program (Part 2) and should give:

```
Python 3.6.3 (v3.6.3:2c5fed86e0, Oct  3 2017, 00:32:08)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>>
 RESTART: /Users/vida/Dropbox/courses/python-iti1120-2018/assignments/assignment3/HW3_solution_GAME.py
Welcome to Single Player Rummy with Dice and strange deck.

The standard deck  has 52 cards: 13 ranks times 4 suits.
Would you like to change the number of cards by changing the number of ranks? ljasd
You are playing with a deck of 52 cards
Here is your starting hand printed in two ways:

104 306 309 313 401 402 408

401 402 104 306 408 309 313
Welcome to round 1.
Please roll the dice.
You rolled the dice and it shows: 2
Since your rolled, 2 the following 2 or 45 (if the deck has less than n 2 cards will be added to your hand from the top

Here is your new hand printed in two ways:

104 304 306 307 309 313 401 402 408

401 402 104 304 306 307 408 309 313
```

```
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind?
Traceback (most recent call last):
  File "/Users/vida/Dropbox/courses/python-iti1120-2018/assignments/assignment3/HW3_solution_GAME.py", line 369, in <mod
    rolled_nonone_round(player)
  File "/Users/vida/Dropbox/courses/python-iti1120-2018/assignments/assignment3/HW3_solution_GAME.py", line 304, in roll
    answer=input("Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind?
KeyboardInterrupt
>>>
 RESTART: /Users/vida/Dropbox/courses/python-iti1120-2018/assignments/assignment3/HW3_solution_GAME.py
Welcome to Single Player Rummy with Dice and strange deck.

The standard deck  has 52 cards: 13 ranks times 4 suits.
Would you like to change the number of cards by changing the number of ranks? yes
Enter a number between 3 and 99, for  the number  of ranks: 3
You are playing with a deck of 12 cards
Here is your starting hand printed in two ways:

101 103 201 202 301 303 403

101 201 301 202 103 303 403
Welcome to round 1.
Please roll the dice.
You rolled the dice and it shows: 2
Since your rolled, 2 the following 2 or 5 (if the deck has less than n 2 cards will be added to your hand from the top o

Here is your new hand printed in two ways:

101 103 201 202 203 301 303 402 403

101 201 301 202 402 103 203 303 403
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? lajsd
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? lajsd
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? 12 13
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 201 202
Invalid sequence. Discardable sequence needs to have at least 3 cards.
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? 201 202 204
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 201 202 204
204 not in your hand. Invalid input
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? 201 202 203
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 201 202 203

Here is your new hand printed in two ways:

101 103 301 303 402 403

101 301 402 103 303 403
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? 101
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 101
Invalid input. Discardable set needs to have at least 2 cards.
Invalid sequence. Discardable sequence needs to have at least 3 cards.
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 104
104 not in your hand. Invalid input
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 101 301

Here is your new hand printed in two ways:

103 303 402 403
```

```
402 103 303 403
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? no
Round 1 completed.
Welcome to round 2.
Please roll the dice.
You rolled the dice and it shows: 1
Discard any card of your choosing.
Which card would you like to discard? 404
404
No such card in the deck. Try again.
Which card would you like to discard? 402

Here is your new hand printed in two ways:

103 303 403

103 303 403
Round 2 completed.
Welcome to round 3.
Please roll the dice.
You rolled the dice and it shows: 3
Since your rolled, 3 the following 3 or 3 (if the deck has less than n 3 cards will be added to your hand from the top o

Here is your new hand printed in two ways:

102 103 302 303 401 403

401 102 302 103 303 403
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 103 303 403

Here is your new hand printed in two ways:

102 302 401

401 102 302
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? yes
Which 3+ sequence or 2+ of a kind would you like to discard? Type in cards separated by space: 102 302

Here is your new hand printed in two ways:

401

401
Yes or no, do you  have a sequences of three or more cards of the same suit or two or more of a kind? no
Round 3 completed.
Welcome to round 4.
The game is in empty deck phase.
Discard any card of your choosing.
Which card would you like to discard? 403
403
No such card in the deck. Try again.
Which card would you like to discard? 401

Here is your new hand printed in two ways:




Round 4 completed.
Congratulations, you completed the game in 4 rounds.
>>>
```