



Programming Assignment 04

-

For loops

Instructions

This programming assignment consists of **2 programming exercises**.

You have to:

1. **create** Python files on your computer (be careful of file names)
2. **edit** them according to the assignment
3. **verify** on your computer that it works (run them and check the output in the shell)
4. **upload** the files to Gradescope (upload directly the **.py** files, **not** a **.zip** file)
5. **check** the autograder report on Gradescope
6. **go back to step 2** if necessary

The autograder will evaluate your code for a few testcases. If some testcases fail, the autograder should show you what is your code output, and what is the expected output.

The autograder will give you a score based on the testcases, but a grader will manually evaluate your coding style after the deadline. Style represents 30% of the coding assignment grade.

Exercise 1 - Word statistics

Write a program (in the file `exercise1.py`) that does the following:

1. Ask the user how many words, between 3 and 6, he will enter
2. Ask the question until the user enter a number between 3 and 6
3. Proceed to ask for each word, one at a time: `Word #[number] please >`
4. At the end, your program should **print out**:
 - (a) the shortest word (the most recently entered shortest if there is a tie)
 - (b) the longest word (the most recently entered longest if there is a tie)
 - (c) the average length of all of the words (up to the second decimal point)

Sample examples (the user input is in red, the printed output is in blue, and the prompt is in black):

```
How many words will you enter? > 4
Word #1 please > wow
Word #2 please > oh
Word #3 please > amazing
Word #4 please > program
Shortest: oh
Longest: program
Average Length: 4.75
```

```
How many words will you enter? > 3
Word #1 please > hi
Word #2 please > hi
Word #3 please > hi
Shortest: hi
Longest: hi
Average Length: 2.00
```

```
How many words will you enter? > 5
Word #1 please > hello
Word #2 please > computer
Word #3 please > python
Word #4 please > world
Word #5 please > homework
Shortest: world
Longest: homework
Average Length: 6.40
```

```
How many words will you enter? > 1
Invalid input. Please enter a number between 3 and 6
How many words will you enter? > 7
Invalid input. Please enter a number between 3 and 6
How many words will you enter? > 6
Word #1 please > hi
Word #2 please > hello
Word #3 please > world
Word #4 please > key
Word #5 please > homework
Word #6 please > no
Shortest: no
Longest: homework
Average Length: 4.17
```

Exercise 2 - Blackjack

It is casino night in the Department of Computer Science and I am in charge of the Blackjack table. You must write a program to help me do my job.

Blackjack is a card game played with one or several standard 52-card decks. The value of the cards is calculated regardless of suits or colors. We will only use letters and digits to indicate the different cards: A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K. The object of the game is to reach 21 points or to get as close to 21 points without going higher. The value of cards two through ten is their pip value (2 through 10). Face cards (Jack, Queen, and King) are each worth ten. Aces can be worth one or eleven, whichever is better for the player. A hand's value is the sum of the card values. Players are allowed to draw additional cards to improve their hands.

Each player has some cards in his hand. They must then make a decision of whether to Hit or Stay. Hit means they request an additional card, Stay means they stop with their current total. Players generally try to Hit until it is likely that another card will push them over 21. For example, if a player has a 5 and a 7, there is a relatively low chance that another card would push them over 21 (only J, Q, and K would do so, since $12 + 10 = 22$). On the other hand, if they have a 5, a 6, and a 7, they will likely stay because any card above 3 will push them over 21 points.

The rules are:

- The dealer must Hit if their total is below 17.
- The dealer must Stay as soon as their total is 17 or higher.
- An Ace (A) should be counted as 11 if it puts the dealer between 17 and 21 points. If it puts them over 21, though, it should be counted as 1.

Write a program (in the file `exercise2.py`) that does the following:

1. ask the user to enter his hand
2. then **print out**:
 - (a) "Hit" if the dealer should take another card.
 - (b) "Stay" if the dealer should not take another card.
 - (c) "Bust" if the sum is already over 21.

Sample examples (the user input is in red, the printed output is in blue, and the prompt is in black):

```
Enter your hand: > A39
Hit
```

```
Enter your hand: > A397
Stay
```

```
Enter your hand: > AA37K
Bust
```

```
Enter your hand: > AAAA8
Hit
```

```
Enter your hand: > 102Q
Bust
```

```
Enter your hand: > AAA7
Stay
```