

**University of Houston**  
**Department of Computer Science**  
**Homework 5**  
**COSC 1306: Computer Science and Programming**  
**Fall 2016**  
**Sent on: 11/10/2016**  
**Due: 11/18/2016 11:59 PM**

Type – Individual Work

**1- (100 points)**

For this homework, you need to download the file Homework5.py, complete it per instructions, and send it to blackboard with **one output of your code in .txt format**. (Attach completed Homework5.py and .txt file in one submission). You have up to three submissions on blackboard and TAs will grade only your last submission.

When the user runs this program, he will be asked to enter the name of a text file. This text file is not given to you but you can assume that it is a small text file with up to 500 characters that can easily be kept in memory. Words are separated by a space but may have a delimiter at the end such as ‘,’ ‘.’ ‘?’ ‘!’. Sentences will end with a new line character ‘\n’.

Let’s assume the following is our text file with 3 lines:

```
misspelling is a common mistake, all people make mistakes!  
  
within a field of study, such as computer graphics, some words might  
be more commonly misspelled, such as pixel misspelled as pixle.  
  
sometimes words are purposely misspelled, as a form of slang,  
abbreviations, or in song lyrics, etc.
```

After receiving the name of the text file, the program will show a menu with 5 choices

- 1- Show all
- 2- Show Line
- 3- Replace
- 4- Save
- 5- Exit

The user may choose one of these options. After choosing an option, a specific task will be done and the menu will be shown again (the menu is in the while loop) till the user chooses 5, at which point the program stops.

Your job is to complete three functions that make the program capable of doing the tasks in the above menu. The user enters a number between 1-5 and per this number the program runs different tasks.

1- Show all:

This option shows all the contents of the text file. To do so the program calls Show(filename). You should complete the Show() function. Do not change the parameters of these function! As you can see this function has two parameters, where one of them is optional. When you call it with one parameter; filename, it will print out all the contents of the text file line by line. In this scenario, the default value for the second parameter is -1. The return value of this function is a string.

2- Show line:

By choosing this option the program asks the user to enter a line number and that line will be shown to the user. The index number for the lines starts at 1. For example, in our example if the user enters 1, the following line will be printed at the output:

*misspelling is a common mistake, all people make mistakes!*

By choosing line number 3, the following will be printed at the output:

*sometimes words are purposely misspelled, as a form of slang, abbreviations, or in song lyrics, etc.*

3- Replace:

By choosing this option, the program gets a line number from the user, shows that line to the user like option 2, and then asks the user to enter two words: the first word, a keyword, is a word that if occurs in that line will be replaced with the second word, a target\_word. The first word and the second word could have different length. Example:

```
Please enter the line number? 1
misspelling is a common mistake, all people make mistakes!
```

```
Please enter your keyword? is
```

```
Replace it with: was
```

```
misspelling was a common mistake, all people make mistakes!
```

Note that the keyword *-is-* is a substring of other words in the line:

```
miisspelling is a common miistake, all people make miistakes!
Your code should only replace all occurrences of is as a word and NOT as a substring in other words.
```

Another example:

```
it is too soon to go to walk now.  
Please enter your keyword? to
```

```
Replace it with: for  
it is too soon for go for park now.
```

Again, in the above example we change all the occurrences of *to* to *for* but *too* was not changed.

After choosing the line number, the line will be fetched by `Show (filename , int(line_number))` and the returned value will be stored in *string*. *string* along with the keyword (*to* in the last example) and the target\_word (*for* in the last example) will be send to `Replace_word(string , keyword , target_word)`. Your job here is to complete this function. The return value of this function should be a string. Remember that in the file, the words are separated by space ' ' and the lines will end with a new line character. For simplicity, you can assume that all letters in the text file are lower cases and all the words input by the user are also in lower case.

#### 4- Save

You should create a new file here and copy the contents of the first file to it. If the user chose option 3 and replaced a word in a line, we have the number of that line in `line_number` and the value of this line in `new_line`. So we apply this change in the `new_line`. `save(filename , new_file_name , new_line , line_number = -1)` complete `save()` function in the following way:

- 1- Open `filename` in read mode and copy all the contents of that to a string variable; let's call it `file_content`.
- 2- If `line_number = -1`, this means that the user never chose option 3. Otherwise, you should change one line in `file_content` (The last change that occurred by running option 3); the number of that line is in `line_number` and the value of this line is in `new_line`
  - Remember that the user chooses the index of lines from 1 but Python starts it from 0. So the actual line that you need to change is `line_number - 1`.
  - In other word, if `line_number = -1` the contents of `filename` and `file_content` will be exactly the same otherwise the difference will be only in one line.
- 3- Open `new_file_name` in the write mode and write the contents of `file_content` in it. Close the `new_file_name`. Print the *success* message!
  - This function will not return any value

- Your original file (filename) will not be changed at any point in your program. You might run option 3 several times but only the result of the last change will be appeared in the new file (new\_file\_name).

#### 5- Exit

If the user chooses option 5, the program will end (the code is already provided for this part).

#### **Example of one run:**

The following is the content of test.txt

```
misspelling is a common mistake, all people make mistakes!  
  
within a field of study, such as computer graphics, some words  
might be more common for misspelling, such as pixel misspelled  
as pixle.  
  
sometimes words are purposely misspelled, as a form of slang,  
abbreviations, or in song lyrics, etc.
```

Please enter your file name: test

```
1-Show all  
2-Show Line  
3-Replace  
4-Save  
5-Exit
```

Please enter your option:1

```
misspelling is a common mistake, all people make mistakes!  
  
within a field of study, such as computer graphics, some words  
might be more common for misspelling, such as pixel misspelled as  
pixle.  
  
sometimes words are purposely misspelled, as a form of slang,  
abbreviations, or in song lyrics, etc.
```

1-Show all  
2-Show Line  
3-Replace  
4-Save  
5-Exit

Please enter your option:2

Please enter the line number? 1  
Misspelling is a common mistake, all people make mistakes!

1-Show all  
2-Show Line  
3-Replace  
4-Save  
5-Exit

Please enter your option: 3

Please enter the line number? 2  
within a field of study, such as computer graphics,

Please enter your keyword? graphics

Replace it with: science  
within a field of study, such as computer science, some words  
might be more common for misspelling, such as pixel misspelled as  
pixle.

1-Show all  
2-Show Line  
3-Replace  
4-Save  
5-Exit

Please enter your option: 3

Please enter the line number? 1  
misspelling is a common mistake, all people make mistakes!

Please enter your keyword? is

Replace it with: was

Misspelling was a common mistake, all people make mistakes!

- 1-Show all
- 2-Show Line
- 3-Replace
- 4-Save
- 5-Exit

Please enter your option:1

misspelling is a common mistake, all people make mistakes!

within a field of study, such as computer graphics, some words might be more common for misspelling, such as pixel misspelled as pixle.

sometimes words are purposely misspelled, as a form of slang, abbreviations, or in song lyrics, etc.

- 1-Show all
- 2-Show Line
- 3-Replace
- 4-Save
- 5-Exit

Please enter your option: 4

Please enter your new file name: result  
Success!

- 1-Show all
- 2-Show Line
- 3-Replace
- 4-Save
- 5-Exit

Please enter your option:1

misspelling is a common mistake, all people make mistakes!

within a field of study, such as computer graphics, some words might be more common for misspelling, such as pixel misspelled as pixle.

sometimes words are purposely misspelled, as a form of slang, abbreviations, or in song lyrics, etc.

1-Show all  
2-Show Line  
3-Replace  
4-Save  
5-Exit

Please enter your option: 5

test.txt

misspelling is a common mistake, all people make mistakes!

within a field of study, such as computer graphics, some words might be more common for misspelling, such as pixel misspelled as pixle.

sometimes words are purposely misspelled, as a form of slang, abbreviations, or in song lyrics, etc.

result.txt

misspelling **was** a common mistake, all people make mistakes!

within a field of study, such as computer **science**, some words might be more common for misspelling, such as pixel misspelled as pixle.

sometimes words are purposely misspelled, as a form of slang, abbreviations, or in song lyrics, etc.

### **Important Considerations for this homework:**

- Do not to import any additional libraries.
- Do not send us your text file that you test your code with! Your code should work with any text file with the mentioned condition in the instruction.
- Use the loops (while and for) in your code. In no point of your code you need to copy-paste part of the code several times. That is why loops are part of every programming language.

### **Instruction for sending your homework:**

- Python 3.0 is the only language that you should use for this homework. **We only accept a .py file for the code and a .txt for one run of the code.**
- Read the instructions of the homework carefully. Do not add anything that was not asked in the instruction. If you are not sure about something, ask TAs by email (allow 24 hours before resending your questions) or attending office hours.
- Start early and send your homework on-time. Submitting homework a few minutes before the deadline is not a good idea. Some students found Blackboard unresponsive in the last minutes before the deadline for homework2. To avoid that, send your homework at least a few hours before the deadline. If you have other difficulties regarding Blackboard you should ask the UH Blackboard Support: <http://www.uh.edu/blackboard/support/>. TAs do not have the student view of the blackboard and cannot help you.
- Late submission penalty is 10 points per day after the deadline. Send your late assignment to [f.pisheh@gmail.com](mailto:f.pisheh@gmail.com) with the title as: "1306\_Homework5\_Lastname\_Firstname\_PSID\_SectionNumber". If you are registered in 11:30-1 class your SectionNumber is 30455 and if you are registered in 1-2:30 class your SectionNumber is 27510.

**Good Luck!**