CSC120H Lab 5

1 Objectives

- Practice using list methods.
- Practice using for loops over lists.

2 Driver and navigator

As always, in this lab you may work on your own or with a partner. You and your partner will take on distinct roles: driver (the person typing at the keyboard) and navigator (the person watching for mistakes, and thinking ahead). You should switch roles after every question below.

3 List Operators and Methods

Complete the tasks below using Python’s operators or list methods in the shell. Use dir(list) and then help, either on list or on a specific list method, to get help on the various list methods that exist.

Please do not submit any code you write in the Shell on MarkUs. This is for your practice and as a warm-up for the sections that follow. But please do NOT skip this section. If you are unsure about anything in this section, please ask your lab TA.

In all cases, assume list L refers to ['Programming', 'is', 'fun'].

1. Write an expression to calculate the length of list L.

2. Write an expression to calculate the number of characters in the first element of the list L.

3. Write an expression that evaluates to True if string 'is' is an element of list L, and to False otherwise.

4. Assume we have executed the following two lines of code:

   ```python
   val = L.extend(L)
   result = L.count('is')
   ```

   • What would print(result) display?
   • What would print(L) display?
   • What does variable val refer to?

5. Given a list L and a value v, write an expression that removes the first occurrence of v from L. You can assume this value is present in the list.

6. Write an expression that adds the string "How are you?" to the front of the list ["I am fine, thank you"]. The list will now have a length of two.
4 Practice For-Loops over Lists

In this section, you will trace through the execution of a code snippet by hand. You will not need to submit anything on MarkUs, but please do NOT skip this part. If you are working in a group, switch driver and navigator.

```python
my_list = [10, 2, 2.45, 1.5]
total = 0
for num in my_list:
    total = total + num
    a = str(num * 2)  # This statement does not serve a particular purpose
    # other than to test your understanding
print('total is', total)
```

Specify, in the table below, what each variable refers to in the beginning and in the end of every iteration of the for-loop. There are four variables: my_list, total, num, and a. We have filled in the value variable total refers to in the beginning of the first iteration. Do the rest.

Beginning of iteration means we are about to execute line with code `total = total + num` but have not executed it just yet. End of iteration means we have just executed line with code `a = str(num * 2)`.

<table>
<thead>
<tr>
<th>#</th>
<th>(i) Beginning of iteration</th>
<th>(ii) End of iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>total -&gt; 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2
5 More Practice with For Loops

In a file called `lab5.py` write the functions described below. You will need to submit that file on MarkUs. Your functions should include at least two examples in the docstring. If you are working with a lab partner, switch driver and navigator for each function.

1. Write a function `sum_list` that takes a list of str and returns the numerical sum of all elements in the list that only contain digits. For example, `sum_list(['hello3', 'csc', '120', 'year', '2017'])` should return 2137 which is the sum of 120 and 2017. The function should return 0 if none of its strings contain only digits. The type contract of this function is `(list of str) -> int`.

2. Write a function `count_even` that takes a list of int and returns the number of elements that are even. Consider 0 an even number. The type contract of this function is `(list of int) -> int`.

6 Declaring Lab Partnership on MarkUs and Submitting

As in the previous labs, you will need to declare, on MarkUs, whether you are working alone or with a partner before you are allowed to submit. Note that you will need to repeat this process for every lab; this is to allow you to switch lab partners or work alone if you choose to.

Once your partnership has been declared, you are ready to submit your work. Although only one partner should submit the assignment, because you declared your partnership, we will know that both of you should get credit for the work – no matter who submits it.

For this lab, you need to submit all the code you wrote for Section 5 in a file named `lab5.py` to MarkUs by Friday October 13th at 9:59pm. Your function names must be exactly as written above. Recall that Python is a case sensitive language. Your functions should not contain any print statements. You should have a single main clause with a call to doctest. All your docstring examples should pass doctest.

Please do NOT add any code/message etc. outside your function definitions, the import doctest, and the main block. Also, you need to ensure that you can run your code before you submit it on MarkUs.