

Sequences and Dictionaries worksheet CS51 - Spring 2026

Practice Problem 1 - Strings

You are given the string `message = 'I love cs51!'`. What will the following expressions evaluate to?

a. `message[1:4]`

b. `message[:2]`

c. `message[2:]`

d. `message[:]`

e. `message[2:11:2]`

f. `message[:11:2]`

g. `message[3::2]`

h. `message[-1]`

i. `message[-1]='0'`

j. `message.upper()`

k. `message.isupper()`

Practice Problem 2 - Lists

a. Define a list `favorite_numbers` with two elements, the numbers 24 and 47.

b. Create an empty list. There are two ways you can achieve this.

c. Convert the string `'cs51'` into a list of its individual characters.

- d. Consider the lists `list_1 = [1, 2]` and `list_2 = [3, 4]`. What will `list_1 + list_2` return?
- e. Consider the list `list_1 = [1, 2]`. What will the contents of `list_2` be if `list_2 = list_1 * 3`?
- f. Consider the list `cheeses = ['Feta', 'Cheddar', 'Edam', 'Gouda']`. What will happen after the execution of each of the following lines?
- ```
cheeses.append('Haloumi')
cheeses.extend(['Emmental', 'Gruyere'])
cheeses.insert(1, 'Brie')
```
- g. Consider the list `cheeses = ['Feta', 'Cheddar', 'Edam', 'Gouda']`. What will happen after the execution of each of the following lines?
- ```
removed_cheese = cheeses.pop(1)
cheeses.remove('Cheddar')
```
- h. How can you break the string `motto = 'cs51 is my favorite class'` into a list of individual words?
- i. How can you concatenate a list of strings into a single string? E.g., how can you turn the list `favorite_list = ['cs51', 'is', 'my', 'favorite', 'class']` into `'cs51 is my favorite class'`?

Practice Problem 3 - Tuples

- a. Define a tuple `tup` with the characters `'c', 's', '5', '1'`.
- b. Create an empty tuple. There are two ways you can achieve this.
- c. Convert the string `'cs51'` into a tuple of its individual characters.
- d. Consider the code `tuple('lup') + ('i', 'n')`. What will happen?
- e. Consider the code `tuple('spam') * 2`. What will happen?
- f. Consider the code `tuple('cs51')`. What will happen if we execute `tuple[3]='0'`?

Practice Problem 4 - Ranges

a. What will the following code do?

```
for i in range(1,10,2):
    print(i)
```

Practice Problem 5 - Dictionaries

a. Create an empty dictionary `d`. There are two ways you can achieve this

b. Create a dictionary that associates English words with their translation to a language of your choice. Insert two such pairs of words. Print in pairs the keys and associated values.

Practice Problem 6 - Sorting strings

```
def compare_class(class_number):
    if class_number < 'cs51':
        print(class_number, 'comes before cs51.')
    elif class_number > 'cs51':
        print(class_number, 'comes after s51.')
    else:
        print('Hello, cs51')
```

What will happen if I call `compare_class('cs50')` and next `compare_class('CS62')`?

Practice Problem 7 - String functions

Define a function `str_odd_indices` that takes one parameter `s` (a string) and returns a string comprised of only the odd indexed characters of `s`. For example, `str_odd_indices('hello!')` would return `'e!'`.

Practice Problem 8 - Aliasing

You are given the function

```
def augment_twice(a_list, val):
    a_list.append(val)
    a_list.append(val)
```

What happens to `numbers` after executing the following code?

```
numbers = [1,2,3]
augment_twice(numbers, 47)
```

What if we were to change the function to this and execute the same code?

```
def augment_twice(a_list, val):  
    a_list = a_list + [val, val]
```

Practice Problem 9 - Recursion

Write a recursive Python function called `rec_linear_search` that takes as input a list, an element, and an index that has a default value of 0, and returns the index of the first encounter of the element in the list, if it exists, or -1 if it does not.

Practice Problem 10 - Loop invariants

Given a list `my_list` (of size `n`) of numbers, write an iterative function `sum_of_list_numbers` that calculates the sum of the numbers in `my_list`.

1. What are the pre-and post-conditions?
2. What is a good loop invariant?
3. Use loop invariants to prove that your function works correctly.