



Columbia College
Vancouver, Canada

**Introduction to Computer Science and Programming 1
CSCI120**

Sample Final Exam

Note: This document has been designed and developed as part of an initiative for creating an OER (Open Education Resource) package for the course CSCI 120 at Columbia College.

Please contact Alireza.davoodi@gmail.com for any comment, modification, and questions.

Terms of use: Please feel free to customize this document as needed

Last Modified: July 2022



Problem1: Analysis/Coding

Suppose a list of numbers is given. For instance, the following list (let's call it *numbers*):

numbers=

1	11	15	13	12	9	8	7	17	14	3	5	6	10	16	18	19	4	2	0
---	----	----	----	----	---	---	---	----	----	---	---	---	----	----	----	----	---	---	---

In this list:

- Each number is repeated only once (no repeated numbers)
- There are 20 items in the list
- The list contains numbers between 0 and 19. (0 and 19 also included)

Also, you been given another list which is: **indexes = [3,6,10,11,17]**

The indexes list represents a list of indexes.

Remember: The items in a list (for instance *numbers*) are indexed from 0.

Now we want to remove all items in numbers list whose indexes are in the list of indexes and return the resulting list. In the resulting list the order of items should be the same as the order of items in the numbers list. The resulting list has all items of the numbers list in the same order excluding the removed items from the numbers list.

Three potential solutions for this problem have been suggested:

Potential Solution1: Create an empty list and using a loop, go through the list of *numbers* and only add items from the *numbers* list to the empty list whose indexes are not in the list of indexes and return the resulting list.

Potential Solution2: Create an empty set and using a loop only add items from the list of numbers to the set whose indexes are not in the list of indexes. Then somehow convert the set to a list in which the order of the items is similar to the order of the items in numbers list.

Potential Solution3: Using the list of numbers above, create a dictionary whose keys are the indexes from 0 to 19 and the values are the corresponding items in the list of numbers. Then remove all pairs from the dictionary whose keys are in the list of *indexes*. Then somehow create a list from the resulting dictionary and return it. Remember the order of the items in the resulting list should be the same as the order of the number in the numbers list.



Now answer the following questions:

Question1.1: Which of the mentioned potential solution(s) could result in a correct answer? (Hint: More than one solution is correct)

Question1.2: Write a python function for every potential solution that you think will generate the correct result. The function should receive two input parameters: numbers and indexes and return a list which contains all items from the numbers whose indexes are not in the indexes list (as described above)

Question1.3: What is the time complexity (order of the complexity) of the solution(s) that you have implemented in Question 2.



Problem2: Analysis

Suppose the following list is given:

1	11	15	13	12	9	8	11	17	14	3	5	9	10	12	11	10	4	2	0
---	----	----	----	----	---	---	----	----	----	---	---	---	----	----	----	----	---	---	---

Question2.1: If you want to use the Binary Search to check whether number 14 is in the list or not, how many times you need to split the list to half in order to find the answer? Visually show it here. No need to write code.

Question2.2: If you want to check whether number 21 is in the list of not using a Linear Search, how many comparisons do you need to make to find the answer. Just write a number and explain why?



Problem3: Design and Programming

Using the simple TV remote shown in the following picture a user should be able to perform the following tasks:

- Increase and decrease the current volume of the TV
- Change the current channel to the previous and next channel.
- Turn on and off the TV
- Mute or unmute the TV

Now: Design an object-oriented model for the remote. To this end, do the following:

- 1- Design a class
- 2- Identify 3 instance variables(parameters) for the class.
- 3- Design a constructor or initializer for the class based on the instance you defined.
- 4- Create an instance method called volumeUp which increase the volume by 1.
- 5- Create an instance method called channelDown which change the channel to the previous channel.
- 6- Create one object of this class.





Problem4: Recursion

The table below shows the list of courses offered in the department and number of students who have registered for the course.

	Course, # of Registered Student
1	CSCI101, 10
2	CSCI105, 12
3	CSCI120, 20
4	CSCI120Flex, 17
5	CSCI125, 23
6	CSCI125Flex, 18
7	CSCI210, 16
8	CSCI225, 22
9	CSCI240, 15
10	CSCI245Flex, 17

Design and implement (Python code) a recursive function to calculate the total number of students registered in all courses.



Problem5: Short Programming

Answer the following questions:

Question1: Convert the following recursive function to a non-recursive one that generates the same result.

```
def function(word):  
    if len(word)==0 or len(word)==1:  
        return word  
    return word[1:]+word[0]
```


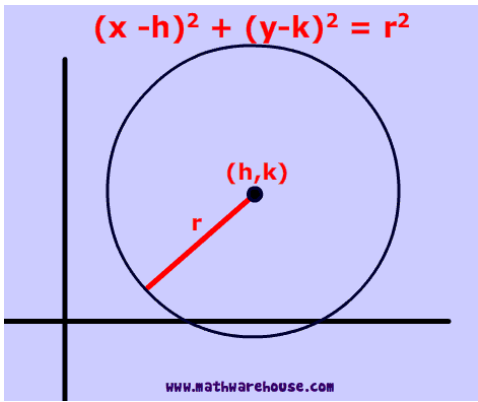
Question2: The following function is supposed to receive a list of numbers and a number as the input parameters. The function finds all occurrences of number inside the list and removes them from the list and eventually returns the results. Do you think this function will work as expected? If yes, please mention and if no, please fix it.

```
def function(listOfNumbers, number):  
    for i in range(len(listOfNumbers)):  
        if listOfNumbers[i]==number:  
            listOfNumbers.remove(number)  
    return listOfNumbers  
  
newList = function([3,4,1,3,4,7,8,1,2,3,4,6,8], 3)  
print("After removing the occurrences of the number, the result is %g" %( newList))
```



Problem6: Defining Variables

- What data type do you use to store the following information:
- Example: a student's first name: We can use
-

Information	Variable – Data type
Example: A student's first name	String
A course's grade	?
The status of TV (whether it is ON or OFF)	?
A product's barcode: 	?
A Circle's equation: A circle is represented using 3 parameters (h,k and r) 	?



Problem7: Programming

- Create a function called, convert. This function receives a string parameter called word which only contains digits (the string represents a positive number) and returns a list of numbers. This is how the function works:
- This function calculates the number of times each digit has repeated in the input string and then generates a number based on that using the following formula and adds it to a list. For instance, if the digit x has been repeated n times, then the function will calculate $n*10+x$ and adds it to the list. (See the example)

Example input: "6743672316"

In the above string:

- 6 is repeated 3 times. Then the corresponding number to be added to the list is $3*10+6 = 36$
- 7 is repeated 2 times: The number to be added to the list $2*10+7 = 27$
- 4 is repeated once: The number to be added to the list is $1*10+4 = 14$
- 3 is repeated 2 times: The number to be added to the list is $2*10+3 = 23$
- 2 is repeated once: The number to be added to the list is $1*10+2 = 12$
- 1 is repeated once: The number to be added to the list is $1*10+1 = 11$

Then the functions *convert* returns the following list: [36,27,14,23,12,11]

Note: The order of the numbers in the list is not important.

Good Luck 😊