



Columbia College
Vancouver, Canada

Introduction to Computer Science and Programming 1
CSCI120

Chapter14: Data Structures and Algorithms

Assignment 14

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



If it is a group assignment, please add the information here

# of Students in the Group:		
Student 1	<i>First name, last name</i>	<i>Student-ID</i>
Student 2	<i>First name, last name</i>	<i>Student-ID</i>
Student 3	<i>First name, last name</i>	<i>Student-ID</i>
Student 4	<i>First name, last name</i>	<i>Student-ID</i>

Requirements

- For the most of the questions in this chapter, you can write the answers in this word document.
- Whenever (if) needed to write code, please add them to a Python file, not on this document.
- Please refer to the Data Structure lecture note if you need help with the questions.



Problem1

- **Problem1.1:** Write a function which receives a list of integers and a number and does liner search to find the number. The function returns True if it finds the number, otherwise it will return False.
- **Problem1.2:** Write a function which receives a list of integers and a number and does a binary search to find the number. The function returns if it finds the number, otherwise it will return False.
- **Problem1.3:** Slightly change your implementation in problem 1.1 and problem 1.2 to count any time your algorithm makes a comparison. How many comparisons is made in each of the above algorithms (Linear search and binary search)
- **Problem1.4:** Try the Problem 1.3 for several lists with different sizes and compare the results.

Problem2

- **Problem2.1:** Write a function which receives a list of integers and does bubble sort to sort the list The function returns the sorted list.
- **Problem2.2:** Write a function which receives a list of integers and does merge sort to sort the list The function returns the sorted list.
- **Problem2.3:** Slightly change your implementation in problem 2.1 and problem 2.2 to count any time your algorithm makes a comparison. How many comparisons is made in each of the above algorithms (Bubble sort and Merge sort)
- **Problem2.4:** Try the Problem 2.3 for several lists with different sizes and compare the results.

Problem3

For each of the following scenarios, specify what data structure would you use to implement the scenario.

1. People staying in the line to submit their passport applications
2. You need to show the results of a live car racing event on the TV. At each time the result shows the rank of participants.
3. You are putting your books in a box. When you add a book to the box you do not touch it anymore.



Problem4

Define a class and its corresponding properties and methods. The class has a property called parenthesis which is a string containing characters including parenthesis. The string is called a MatchingParenthesis string if any open parenthesis is matched with a close one. For instance, the followings are MatchingParenthesis:

- ()
- (())
- ((()))
- (s(aaa))
- (3(#(aa)1))

The followings are examples of Non-MatchingParenthesis strings

- (
- (()
- (())
- (4(5)0))7

Complete the class you defined above with methods to check whether a string is a MatchingParenthesis one or not.

What data structure would you use to solve this problem?

What is the time complexity of your solution.

Good Luck ☺