Data Structure - Sorting and Searching

Question 1 (25 points)

Using Microsoft Word or any text editors, conduct research and write a paper (with a minimum of 150 words) to answer the following question. Please include references using the APA format citation guide.

Describe what is sorting and searching, and why are they essential in a computer science field. Give two examples when sorting and searching are necessary for designing software applications. Describe three different types of existing sorting algorithms and two types of searching algorithms. Justify, compare and contrast your choice of sorting and searching algorithms.

Question 2 (60 points)

There are various sorting algorithms available to sort data of different sizes. Three of these algorithms are Bubble sort, Shell sort, and Quicksort. Write a program to generate random integer numbers of multiple sizes; 10000, 30000, 50000, 70000, and 90000, and find out which of these sorting algorithms performs the fastest sorting technique. You can write the sorting program yourself or use an existing sorting program that you can find on the Internet and modify them to fit your need. Provide data to prove and support your findings by *plotting a line graph* showing the time each takes to sort data of various sizes. Please answer what can you conclude from your data visualization chart.

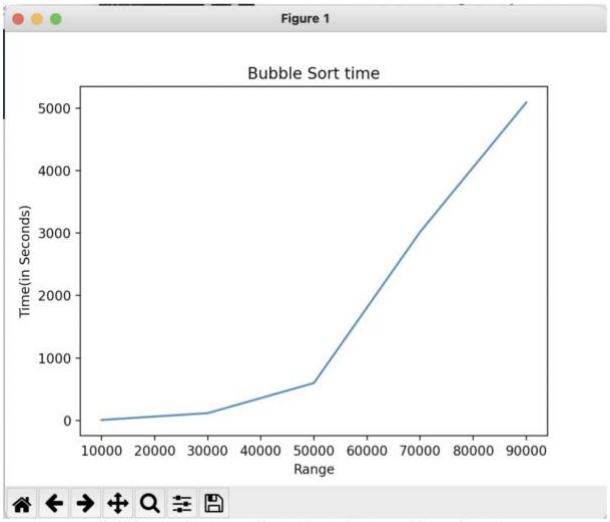
Please note that your program will take time to complete the execution process due to the larger dataset handling. The lesson here is to be aware of how significant it is to select the appropriate algorithm to complete a task to save time. Time is a luxury you and I do not have!

Below is a sample code on how to time the sorting duration.

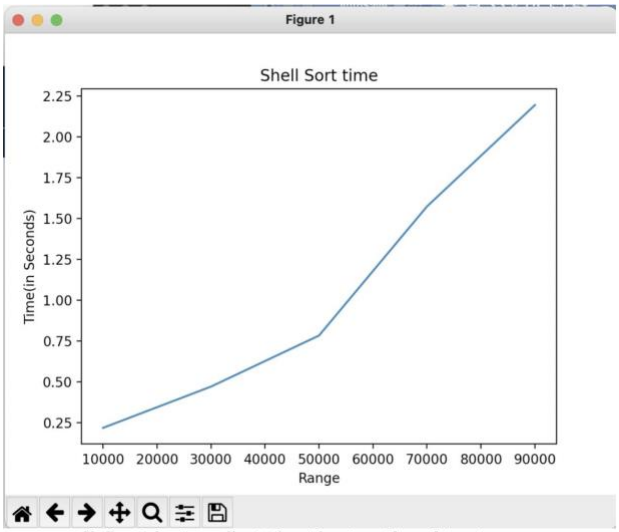
```
import time
import random

mylist = []
for i in range(0,10000): # create 10000 random numbers
    x = random.randint(1,10000) # ranges 1 to 10000
    mylist.append(x)

start_time = time.time() # start the clock
sorting_method_function(mylist) #  ← put your sorting method
stop_time = time.time() - start_time # stop the clock and calculate the time difference
print("--- the sort tool %s seconds ---" % stop_time)
```

NOTE: Everything should be written in one program. If you wish, you may create a separate file for each sorting program as a function and import them into your main program.

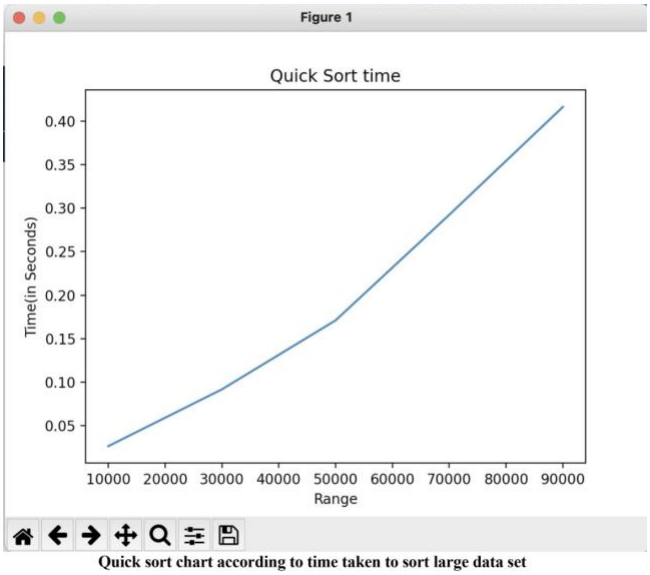
Below is a sample of the result of running the sorting comparison programs. Please be aware that the result depends on the random datasets created by the program. Your result may slightly be different.



Bubble sort chart according to time taken to sort large data set



Shell sort chart according to time taken to sort large data set



Quick sort chart according to time taken to sort large data set

Question 3 (15 points)

Explore on the Internet what can Python package ***pysort*** offers Software Developers when writing a program. Describe how to use the existing package and give some programming examples.

Learning Report Summary

Using Microsoft Word or any other text editor, write a summary report explaining how you completed your assignment and what you learned from completing your assignment.

1. Did you successfully get your assignment done? Did it run? Any error? Did you get the correct result? Did you test your program thoroughly?
2. How much time did you spend completing your assignment?
3. Did you find the assignment easy or challenging for you?
4. Did you write the program yourself? Did you get any help from anyone?
5. How did you resolve the issues when you encountered obstacles to completing your program? Did you use Google to get help? Describe how Google was able or not able to assist you.
6. What did you learn from doing this assignment?
7. Any other information you would like to share with your instructor? Make sure you provide program output on each option.

What to submit

Submit all your work, the programs, and the programs' output.