GTU Department of Computer Engineering

CSE 222/505 - Spring 2022

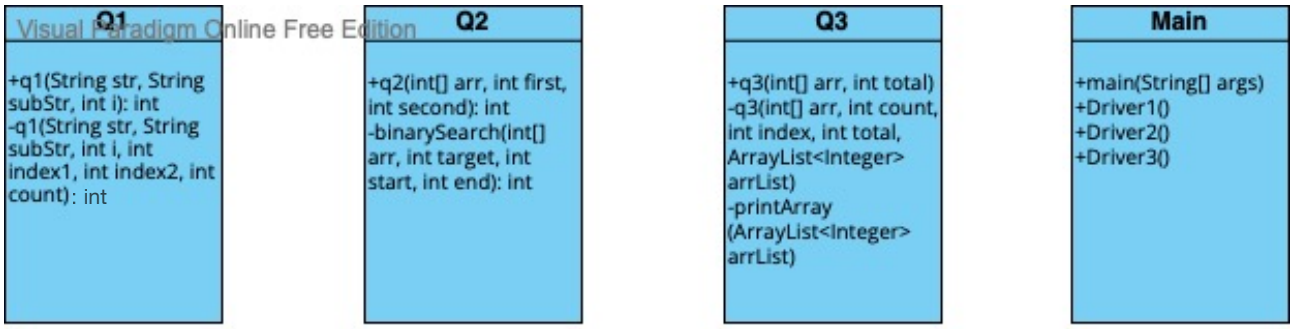
Homework 4 Report

Sena Özbelen
1901042601

1. System Requirements

-All functions have to be called with proper parameters.
-First method should be given two string and an integer indicating the occurrence.
-Second method should be given a sorted integer array and two numbers to get the interval.
-Third method should be given an unsorted integer array and an integer indicating the sum.

2. Class Diagram



3. Problem Solution Approach

In the first question, the method is based on comparing letters individually. If the current index of the main string is equal to the length of the main string, then it returns -1 (base case). Otherwise, it compares the letter of the main string to the letter of the substring. Whenever, a substring is found in the main string, the counter increases. When it finds the ith occurrence, it returns the index of first letter in the main string.

In the second question, the method directly performs two binary search. If the exact numbers are in the array, then it finds their indexes. Otherwise, it finds the closer and higher numbers in the array and binary search locates them. The difference between the indexes gives the number of items between them. The boundary numbers are excluded if they are in the array.

In the third question, the method starts from 0th index and it keeps summing next items until the sum exceeds the total sum. If it reaches the exact total sum, then it prints the items stored in an array list. The method is called recursively with an increase of the index.

The fourth question is explained in Time Complexity part.

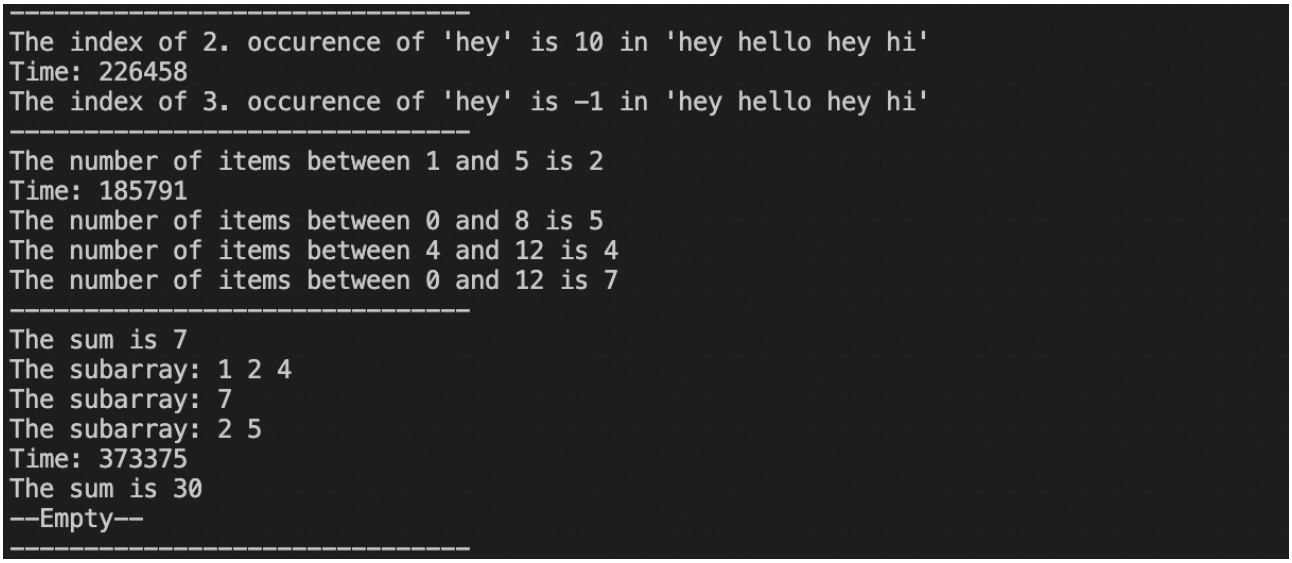
4. Test Cases

For the first question,
-Find 2. occurrence of ‘hey' in "hey hello hey hi”
-Find 3. occurrence of ‘hey' in "hey hello hey hi” (returns -1)

For the second question,
-arr = {1,2,4,5,7,9,11}
-Find the number of items between 0 and 8
-Find the number of items between 4 and 12
-Find the number of items between 0 and 12

For the third question,
-arr = {1,2,4,5,7,2,5}
-Find the subarrays whose items’ sum is 7
-Find the subarrays whose items’ sum is 30

5. Running Command and Results



Time Complexity

1.

$T(n) = T(n-1) + c2$

$T(n) = T(n-k) + k*c2$ and $T(0) = c1$

Where $k = n$,

$T(n) = c1 + n*c2 \Rightarrow O(n) = n$

2.

Binary Search Method:

$T(n) = T(n/2) + c2$

$T(n) = T(n/2^k) + k*c2$ and $T(1) = c1$

Where $k = \log n$,

$T(n) = c1 + \log n*c2 \Rightarrow O(n) = \log n$

Overall:

$O(n) = 2\log n+2n+c3 = n$

3.

$T(n) = T(n-1) + n + c2$

$T(n) = T(n-k) + k*n + k*c2 -(k-1)-(k-2)-\ldots-0$ and $T(0) = c1$

Where $k = n$,

$T(n) = c1 + n^2 + n*c2 - (n-1)(n-2)/2 \Rightarrow O(n) = n^2$

(printArr method is ignored)

4.

This method takes 2 integers and at first, it checks whether one of them is lower than 10. If it is not, then it applies split method. This split method takes 2 parameters, integer itself and half. Half is the half of the max digit of these 2 numbers. Split Integer method divides the number by 2^half(for binary, 10^half for decimal) and stores the result to first integer. After that, it applies modulus operation and stores the result to second integer. There are 3 recursive calls and each call's return is stored in sub0, sub1, sub2, respectively. When all calls are done, the last statement in return is performed. The result is the multiplication of two given numbers based on finding the digits individually.

$T(n) = 2T(n/2) + c3$

-The first recursive call is called only once and number_of_digit is ignored

$T(n) = (2^k)*T(n/2^k) + (2^k-1)*c3$ and $T(1) = c2$

Where $k = \log n$,

$T(n) = n*c2 + (n-1)*c3 \Rightarrow O(n) = n$