Data Structures and Algorithms

COSC 336 Assignment 1

Instructions.

1. Due date and time: As indicated on Blackboard.
2. This is a team assignment. Work in teams of 3-4 students. Submit on Blackboard one assignment per team, with the names of all students making the team.
3. Your programs must be written in Java.
4. Write your programs neatly - imagine yourself grading your program and see if it is easy to read and understand.

   Comment your programs reasonably: there is no need to comment lines like "i++" but do include brief comments describing the main purpose of a specific block of lines.
5. You will submit on **Blackboard** two files.

   The **1-st file** is a pdf file (produced ideally with latex and Overleaf) and it will contain the following:

   (a) The solution to the Exercises.

   (b) A short description of your algorithms for the Programming Tasks 1, where you explain the dynamic programing approach (see the sketch of the **Algorithm** below). More precisely, you need to indicate how you compute $d[0]$ (this is the initialization step), and how you compute for every $i \geq 1$, the value of $d[i]$ using the values of some of the previous $d[j]$'s, for $j < i$).

   (c) A table with the results your program gives for the 4 data sets indicated for the programming task.

   (d) The java code (so that the grader can make observations).

   The **2-nd file** is the .java file containing the java source code for Programming Task 1.

Exercise 1

Consider the following three program fragments (a), (b), and (c).

(a)
```
sum = 0;
for (int i = 0; i<100 * n ; i++) {
        sum++;
}
```

(b)
```
sum = 0;
for (int i = 0; i< 2*n ; i++) {
        sum++;
}
```

(c)
```
sum = 0;  i=n*n;
while (i > 1) {
        sum++;
        i= i/2;
}
```

We denote by $T_a(n), T_b(n), T_c(n)$ the running time of the three fragments.

1. Give Θ evaluations for $T_a(n), T_b(n), T_c(n)$.
2. Is $T_b(n) = O(T_a(n))$? Answer YES or NO and justify your answer.
3. Is $T_c(n) = \Theta(T_a(n))$? Answer YES or NO and justify your answer.

Exercise 2.

Give an example of a function $f(n)$ with the property that $f(n)$ is $\omega(n^2)$ and also $f(n)$ is $o(n^3)$.

Exercise 3. Indicate the running time of the following program fragment in the $\Theta(\cdot)$ notation.

```
x = 0;

for {i=1 ; i <= 2n+3; i++)

    for (j= 1; j < = 3n+7; j++)

        x = x+1
```

Programming Task 1.

Your program will calculate the length of a longest contiguous subsequence of equal values in a given sequence of integer values. For example if the input sequence is 1,0,0,0,1,1 your program will return 3. Another example: 2,5,5,1,11,11,11,3,5,5,5,5,4,7. Now your program should return 4, because there are 4 consecutive values (namely four 5's) and there is no longer subsequence of equal values.

Algorithm. Use dynamic programming. Suppose the initial sequence is $a[0], a[1], \ldots, a[n-1]$. The subproblems are: $d[i]$ = length of the longest contiguous of equal values ending with $a[i]$. Think how to calculate $d[0]$, and next how to calculate $d[i]$ if you know the previous values of $d[\cdot]$.

Test your program on the following sequences and insert in the first file (the pdf file) that you submit tables with the results for each sequence:

- $2, 5, 5, 1, 11, 11, 11, 3, 5, 5, 5, 5, 4, 7$
- $1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0,$
  $0, 0, 0, 0, 0, 1$
- $1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7,$
  $1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2$
- Using the Java pseudo-random number generator, generate a sequence with 4000 bits (so 4000 0s and 1s) and run your program on that sequence.