**CSE 2320** **Name: HOUDA BELEFQIH**

**Homework 4 part a** (40 points)

(See changes in red color, added 3/22, 11:55am)

**Task 1 – 11 points**

a) (5 points) Fill-out the edit distance table for the words **NONSTOP** and **ROUND.** An empty table is provided below. Fill-out however much you need.

| i \ j | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | “” | N | O | N | S | T | O | P | |
| 0 | “” | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| 1 | R | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| 2 | O | 2 | 2 | 1 | 2 | 3 | 4 | 5 | 6 | |
| 3 | U | 3 | 3 | 2 | 2 | 3 | 4 | 5 | 6 | |
| 4 | N | 4 | 3 | 3 | 2 | 3 | 4 | 5 | 6 | |
| 5 | D | 5 | 4 | 4 | 3 | 3 | 4 | 5 | 6 | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |

~~Recover the trace for the following edit distance table~~. Answer questions b), c), d) for the table below. The symbols indicate: \ - diagonal, ^ - arrow pointing up, < - arrow pointing left

| | | r | e | g | r | e | s | s | i | o | n |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | <mark><</mark> | < | < | < | < | < | < | < | < | < | < |
| s | ^ | <mark>\</mark> | \ | \ | \ | \ | \ | \ | < | < | < |
| e | ^ | \ | <mark>\</mark> | < | < | \ | < | \ | \ | \ | \ |
| g | ^ | \ | ^ | <mark>\</mark> | < | < | < | < | < | < | < |
| m | ^ | \ | ^ | ^ | <mark>\</mark> | \ | \ | \ | \ | \ | \ |
| e | ^ | \ | \ | ^ | \ | <mark>\</mark> | <mark><</mark> | <mark><</mark> | <mark><</mark> | < | < |
| n | ^ | \ | ^ | ^ | \ | ^ | \ | \ | \ | <mark>\</mark> | \ |
| t | ^ | \ | ^ | ^ | \ | ^ | \ | \ | \ | \ | <mark>\</mark> |

b) (2 points) Show in the table the path you followed (e.g. bold, highlight, circle).

c) (3 points) Show all 3 strings: the 2 strings that show the word alignments and the 3rd one showing the cost.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| r | e | g | r | e | s | s | i | o | n |
| s | e | g | m | e | - | - | - | n | t |
| **x** | **.** | **.** | **x** | **.** | **x** | **x** | **x** | **x** | **x** |

d) (1 point) Using the 3rd string, what is the edit distance between these 2 strings (or the cost of the string alignment)? **Cost = 7**

**Task 2 – 5 points**

Given an **unlimited number** of the items below:

| Item: | A | B | C | D |
|---|---|---|---|---|
| Weight: | 4 | 6 | 10 | 12 |
| Value: | 10 | 21 | 33 | 36 |

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sol: | **0** | **0** | **0** | **0** | **10** | **10** | **21** | **21** | **21** | **21** | **33** | **33** | **42** |
| Picked | **-** | **-** | **-** | **-** | **A** | **A** | **B** | **B** | **B** | **B** | **C** | **C** | **B** |
| A | - | - | - | - | 0<br>10 | 1<br>10 | 2<br>10 | 3<br>10 | 4<br>20 | 5<br>20 | 6<br>31 | 7<br>31 | 8<br>31 |
| B | - | - | - | - | - | - | 0<br>21 | 1<br>21 | 2<br>21 | 3<br>21 | 4<br>31 | 5<br>31 | 6<br>42 |
| C | - | - | - | - | - | - | - | - | - | - | 0<br>33 | 1<br>33 | 2<br>33 |
| D | - | - | - | - | - | - | - | - | - | - | - | - | 0<br>36 |

For all the programming tasks below write your answer in a C file called hw4_a.c .

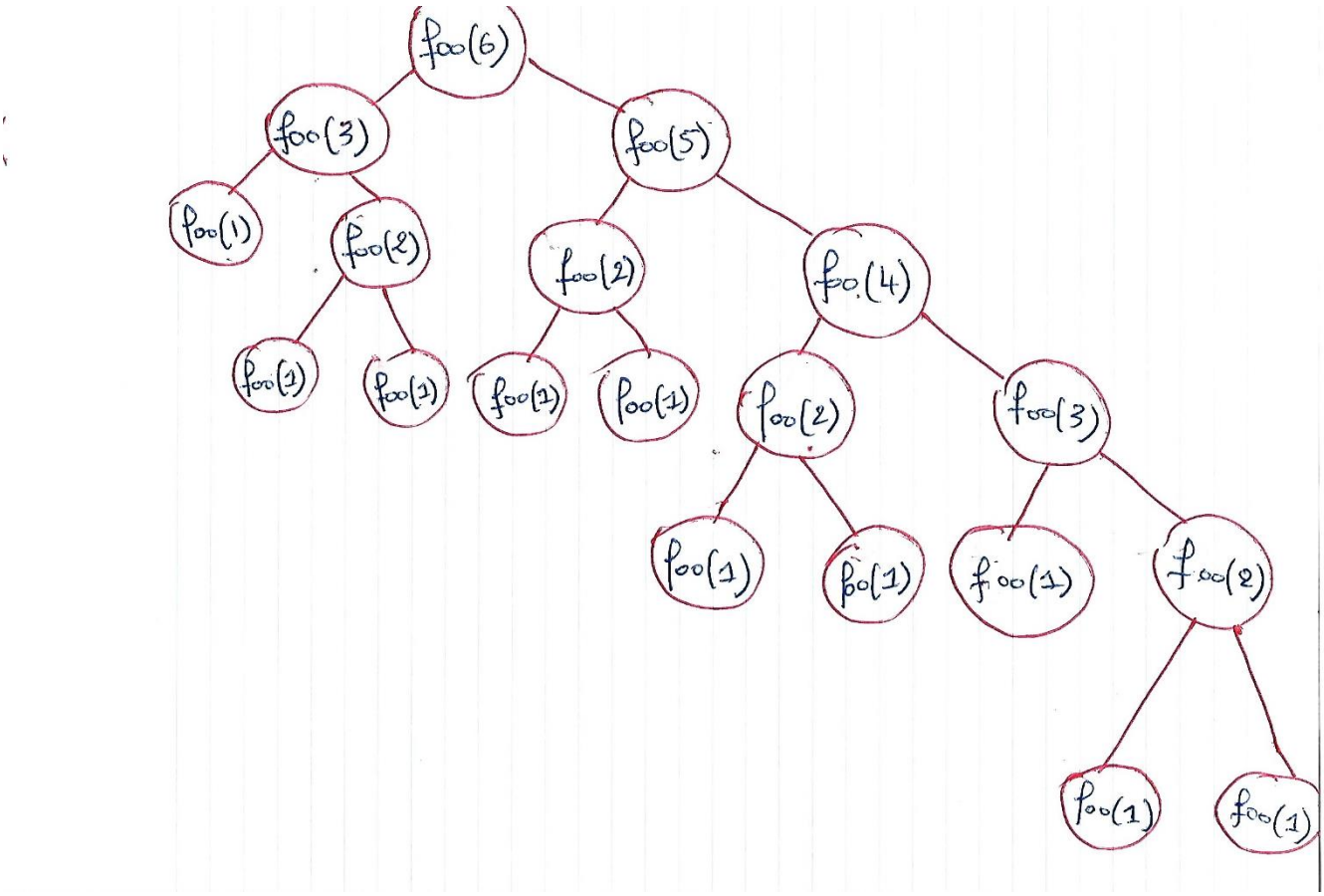
**Task 3 – 12 points** Consider this recursive function **foo(N):**

```
int foo(int N){
    if (N <= 1) return 5;
    int res1 = 3*foo(N/2);
    int res2 =   foo(N-1)
    if (res1 >= res 2)
        return res1;
    else
        return res2;
}
```

a) (3 points) Write the recurrence formula for the TIME COMPLEXITY of this function (including the base cases) for N>=0. (You do NOT need to solve it.) (Note that it should NOT be the formula for what the function computes, but for how long it takes.)

$$T(N) = N*T(N/2) + N*T(N-1) + N^2 + c$$ ***c being a constant***

b) (3 points) Draw the tree that shows the function calls performed in order to compute foo(6) (the root will be foo(6) and it will have a child for each recursive call.)



c) (6 points) Write this code in hw4_a.c.

(4 points)Re-implement this function with memorization (i.e. use a solution array to look-up and store results of recursive calls).

(2 points) In addition to showing the changes in the foo function, **also write the wrapper function** that calls the memorized **foo** function.

**Question 4 – 12 points** Recursion – write code

a) (9 points) Write the code in hw4_a.c
Write a **recursive** function that takes as argument an array and its length and returns 1 if the elements in the array are in order (either increasing or decreasing) and 0 otherwise. The solution must use recursion. You can write auxiliary (recursive) functions if needed. E.g.:
- for [3,6,1,8] it returns 0 (neither increasing nor decreasing).
- for [3, 6, 8, 50, 10000] it returns 1 (increasing).
- for [20,16, 9, 7, 3] it returns 1 (decreasing).
- for [20,6,-1,-5] it returns 1 (decreasing).
- for an empty array **or an array with 1 element** it returns 1.
**- if all the array elements are equal it should return 1. – Updated 3/24/17**

b) (2 points) Write the recursion formula for your function. (Include this answer here or with the code)

c) (1 points) What is the runtime of your function? (No justification needed.) (Include this answer here or with the code)