CS 5800: Algorithms (Spring 2020) **Ahmed Ezzat**
Student: Samuel Steiner Due Jan. 31, 2020

Homework 2

1 Question 1 [1pt]: Chapter-4 (page 74): Exercise 4.1-2

Question:

Write Pseudocode for the brute-force method of solving the maximum sub-array problem. Your procedure should run in $\Theta(n^2)$ time

Answer:

```
brute-force-max-sub-array(A)
    n = A.length
    max = -infinity
    for i = 1 to n
        sum  = 0
        for h =1 to n
            sum = sum + A[h]
            if sum > max
                max = sum
                low = i
                high = h
    return
```

2 Question 2 [2pt]: Chapter-4 (page 108): Exercise4.3

Question:

Give asymptotic upper and lower bounds for T(n) in each of the following recurrences/ Assume that T(n) is constant for sufficiently small n. Make your bounds as tight as possible, and justify your answers:

$$a)T(n) = 4T(n/3) + n\lg n$$
$$b)T(n) = 3T(n/3) + n/\lg n$$
$$c)T(n) = 4T(n/2) + n^2\sqrt{n}$$

Answer:

a) By master theorem: $T(n) = \Theta(n^{\log_3 4})$
b)

$$\begin{aligned} T(n) &= 3T(n/3) + \frac{n}{\lg n} \\ &\le cn\lg n - cn\lg 3 + \frac{n}{\lg n} \\ &= cn\lg n + n(\frac{1}{\lg n} - c\lg 3) \\ &\le cn\lg n \end{aligned}$$

$$\begin{aligned} &\text{for } \varepsilon > 0 \\ T(n)&3T(n/3) + \frac{n}{\lg n} \\ &\ge 3c/3^{1-\varepsilon}n^{1-\varepsilon} + \frac{n}{\lg n} \\ &= 3^{\varepsilon}cn^{1-\varepsilon} + \frac{n}{\lg n} \end{aligned}$$

$$\text{This is the same as showing}$$
$$3^{\varepsilon} + n^{\varepsilon}/c\lg n \ge 1$$

Since $\lg n \in o(n^{\varepsilon})$ this inequality holds. So, we have that The function is soft Theta of n
c) By master theorem: $T(n) = \Theta(n^{2.5})$