Fundamental Algorithms, Home work-1

Tulsi Jain

First Problem

Searching Algorithms(A)

Data: Array A, of size n
Result: Maximum element in an array, A
if *A.length > 0* **then**
  var tempMax = A[1];
  for *j=2 to A.length* **do**
    if *A[j] > tempMax* **then**
      tempMax = A[j];
    else
      do nothing
    end
  end
  return tempMax;
else
  return Array is empty
end

Algorithm 1: Find Maximum Element

Invariant: At any point of time, before each iteration of the for loop variable named **tempMax** is equal to the maximum of in A [1, 2, - - - - -, j-1]
Condition: J $\leq$ A.length

Proof of Correctness

a Initialization

At starting J = 2, hence A = [1, ... j-1] contains only one element i.e. first element tempMax is too, equal to A[1], hence invariant holds in the beginning.

b Maintenance

Assumption:
Invariant holds for j -1 iteration
As per the assumption, **tempMax** is equal to the maximum of in A [1, 2, - - - - -, j -1]. When **for loop**

is executed for jth iteration, then tempMax would get replaced with maximum of amongst tempMax and A[j] as per the below mentioned code

```
if A[j] > tempMax then
| tempMax = A[j];
else

end
```

Hence, invariant also holds during maintaince.

c Termination

At termination, j = n+1, hence hence A = [1, ... j-1] is the complete array i.e A = [1,2, ... n-1, n] And **tempMax** is equal to maximum of in array A. Hence loop invariant holds at the termination too. Moreover, this the purpose of the algorithms too.
Condition is false as j(n+1) is exceeded A.length(n)

Loop invariant is valid at all three steps, before iteration, during iteration and after iteration. Also this loop invariant also shows at the termination tempMax in maximum of in array A (hence useful).

Second Problem

$$L_n = \phi^n + (1-\phi)^n$$

This needs to be prove using induction approach a, where $\phi = \frac{1+\sqrt{5}}{2}, goldenratio$

Solution
Assumption: Let's assume this is true $\forall$ i, i <= n
Now, lets prove it for n +1;

$$L_{n+1} = \phi^{n+1} + (1-\phi)^{n+1}$$

$$L_{n+1} = L_n + L_{n-1}$$

(Given in problem statement)

$$L_n = \phi^n + (1-\phi)^n$$

$$L_{n-1} = \phi^{n-1} + (1-\phi)^{n-1}$$

$$\phi^n + (1-\phi)^n + \phi^{n-1} + (1-\phi)^{n-1} = \phi^{n+1} + (1-\phi)^{n+1}$$

$\phi = \frac{1+\sqrt{5}}{2}$

(Squaring both sides)

(Proofing golden ratio square is equal to golden ratio + 1)
$\phi^2 = \frac{1+5+2\sqrt{5}}{4}\phi^2 = \frac{3+\sqrt{5}}{2}\phi^2 = \frac{1+\sqrt{5}}{2} + 1\phi^2 = \phi + 1$

$$\phi^n + (1-\phi)^n + \phi^{n-1} + (1-\phi)^{n-1} = \phi^{n+1} + (1-\phi)^{n+1} \tag{1}$$

$$\phi^{n-1}(\phi+1) + (1-\phi)^{n-1}(1-\phi+1) = \phi^{n+1} + (1-\phi)^{n+1} \tag{2}$$

take $\phi^{n-1} and$(1- $\phi)^{n-1}$ common

$$\phi^{n-1}(\phi^2) + (1-\phi)^{n-1}(1-\phi+\phi^2-\phi) = \phi^{n+1} + (1-\phi)^{n+1} \tag{3}$$

replace ($\phi + 1) with (\phi^2), , , 1 with \phi^2 - \phi$

$$\phi^{n+1} + (1-\phi)^{n-1}(1-\phi)^2 = \phi^{n+1} + (1-\phi)^{n+1} \tag{4}$$

$$\phi^{n+1} + (1-\phi)^{n+1} = \phi^{n+1} + (1-\phi)^{n+1} \tag{5}$$

Both side are equal

Third Problem

a. Being in Θ is an equivalence relation.

To prove this we need, to all three reflexive, symmetric and transitive properties hold true.

I Transitive

$$if \quad f(n) = \Theta(g(n) \quad And \quad if \quad g(n) = \Theta(h(n), \quad Does \quad it \quad imply? \quad f(n) = \Theta(h(n))$$

$$lim_n \to \infty \frac{f(n)}{g(n)} = \quad C_1 > 0 \tag{1}$$

$$lim_n \to \infty \frac{g(n)}{h(n)} = \quad C_2 > 0 \tag{2}$$

Multiple both 1 and 2

$$lim_n \to \infty \frac{f(n)}{h(n)} = C1 * C2 > 0 \tag{3}$$

Hence Transitive property holds for being in Θ

II Symmetric

$$if \quad f(n) = \Theta(g(n) Does \quad it \quad imply? \quad g(n) = \Theta(f(n))$$

$$lim_n \to \infty \frac{g(n)}{f(n)} = C_1 > 0 \tag{1}$$

Hence Symmetric property holds for being in Θ

III Reflexive

$$if \quad f(n) = \Theta(g(n) Does \quad it \quad imply? \quad f(n) = \Theta(f(n))$$

$$lim_n \to \infty \frac{f(n)}{f(n)} = 1 > 0 \tag{1}$$

Hence Reflexive property holds for being in Θ

All three properties hold true hence equivalence relation holds for Θ.

b. Maximum of two functions is in Θ of their sum

$$Prove, \quad max(f(n), g(n)) = \Theta(f(n) + g(n))$$

Assumption both functions are non-negative

$$lim_n \to \infty \frac{max(f(n), g(n))}{f(n) + g(n)} \tag{1}$$

Let's say highest degree in $g(n) >= f(n)$, then for $lim_n \to \infty$ We can write $lim_n \to \infty \frac{max(f(n),g(n))}{f(n)+g(n)}$ as $lim_n \to \infty \frac{g(n)}{f(n)+g(n)}$. As order of $g(n) >= f(n)$ is by applying L's Hospital Rule. $f(n)$ would become irrelevant. This would not be either 0 or ∞ but in between. Let's say it C_1.

As C_1 lies between 0 and ∞. So statement maximum of two functions is in Θ of their sum is true.

c. Sum of two functions is in Θ of their maximum

$$Prove \quad f(n) + g(n) = \Theta(max(f(n), g(n)))$$

Assumption both functions are non-negative

$$lim_n \to \infty \frac{f(n) + g(n}{max(f(n), g(n))} \tag{1}$$

Let's say highest degree in $g(n) >= f(n)$, then for $lim_n \to \infty$ We can write $lim_n \to \infty \frac{f(n)+g(n)}{max(f(n),g(n))}$ as $lim_n \to \infty \frac{f(n)+g(n)}{g(n}$. As order of $g(n) >= f(n)$ is by applying L's Hospital Rule. $f(n)$ would become irrelevant. This would not be either 0 or ∞ but in between. Let's say it C_1.

As C_1 lies between 0 and ∞. So statement maximum of two functions is in Θ of their sum is true.

Fourth Problem

We can choose either of the five asymptotic notation to show the relation between two functions. Let's choose o notation.

$$f(n) = o(g(n)), \quad \text{if} \quad lim_{n\to\infty}\frac{f(n)}{g(n)} = 0$$

1. Functions: Logarithmic & Constant

Logarithmic order of growth $\mathcal{O}(\log n)$
Constant order of growth $\mathcal{O}(c)$
Logarithmic function grows asymptotically fatser than the Constant function,
c = $o(\log n)\forall c$
compute, $\quad lim_{n\to\infty}\frac{c}{logn}$
$lim_{n\to\infty}\frac{c}{logn} = 0$

2. Functions: Linear & Logarithmic

Linear order of growth $\mathcal{O}(n)$
Logarithmic order of growth $\mathcal{O}(\log n)$
Linear function grows asymptotically fatser than the Logarithmic function,
$\log n$ =o(n)
compute, $\quad lim_{n\to\infty}\frac{\log n}{n}$
Applying L-hospital's rule (differentiating both denominator and numerator w.r.t n)
$lim_{n\to\infty}\frac{\frac{1}{n}}{1} = 0$

3. Functions: Linearithmic & Linear

Linearithmic order of growth $\mathcal{O}(n\log n)$
Linear order of growth $\mathcal{O}(n)$
Linearithmic function grows asymptotically fatser than the Linear function,
n = $o($nlog n
compute, $\quad lim_{n\to\infty}\frac{n}{n\log n}$
Applying L-hospital's rule (differentiating both denominator and nominator w.r.t n)
$lim_{n\to\infty}\frac{1}{n*\frac{1}{n}+\log n*1} = lim_{n\to\infty}\frac{1}{1+\log n} = 0$

4. Functions: Polynomial & Linearithmic

Polynomial order of growth $\mathcal{O}(n^k)$
Linearithmic order of growth $\mathcal{O}(n\log n)$
Polynomial function grows asymptotically fatser than the Linearithmic function,
nlog n =o(n^k) $\quad\forall\quad k > 1$
compute, $\quad lim_{n\to\infty}\frac{n\log n}{n^k}$
Applying L-hospital's rule (differentiating both denominator and nominator w.r.t n)

$lim_{n\to\infty}\frac{n\log n}{n^k} = lim_{n\to\infty}\frac{n*\frac{1}{n}+\log n*1}{k*n^{k-1}} = lim_{n\to\infty}\frac{1+\log n}{k*n^{k-1}} = lim_{n\to\infty}\frac{0+\frac{1}{n}}{k*(k-1)*n^{k-2}} = lim_{n\to\infty}\frac{1}{k*(k-1)*n^{k-1}} = 0 \quad as \quad k > 1$

5. Functions: Exponential & Polynomial

Exponential order of growth $\mathcal{O}(a^n)$
Polynomial order of growth $\mathcal{O}(n^k)$
Exponential function grows asymptotically fatser than the Polynomial function, for a > 0 and k > 0
n^k =o(a^n)
compute, $\quad lim_{n\to\infty}\frac{n^k}{a^n}$
Applying L-hospital's rule (differentiating both denominator and nominator w.r.t n)
$lim_{n\to\infty}\frac{n^k}{a^n} = lim_{n\to\infty}\frac{k*n^{k-1}}{\ln a*a^n} = lim_{n\to\infty}\frac{k*(k-2)*n^{k-1}}{\ln a^2*a^n} = lim_{n\to\infty}\frac{k!}{\ln a^k*a^n} = 0$