CSE222 HW2 200104004019 Resit Aydın

Subject : Date :...../...../.....

(1.)

a) $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \frac{n^2+7n}{n^3+7} = \frac{1}{\infty} = 0$ so $f(n) = O(g(n))$

b) $\lim_{n\to\infty} \frac{12n+\log_2 n^2}{n^2+6n} = \frac{1}{\infty} = 0$ so $f(n) = O(g(n))$

c) $\lim_{n\to\infty} \frac{n \cdot \log_2 5n}{n+\log_2 8n^3} = \infty$ so $f(n) = \Omega(g(n))$

d) $\lim_{n\to\infty} \frac{n^n+5n}{3+2^n} = \infty$ so $f(n) = \Omega(g(n))$

e) $\lim_{n\to\infty} \frac{3^{2n}}{5^{2n}} = \frac{1}{\infty} = 0$ so $f(n) = O(g(n))$

(2)

a) $F(n) = n \cdot O(1) = O(n)$

b) $F(n) = n \cdot O(1) \cdot n \cdot O(1) = n^2 \cdot O(1) = O(n^2)$

c) There occured an infinite loop, so it has no time complexity.

d) $n \cdot O(1) = O(n)$

(3)

Time complexity of withoutLoop method $\rightarrow O(1) \cdot n = O(n)$

Time complexity of withLoop $\rightarrow O(1) \cdot n = O(n)$

So, their time complexity is the same but withLoop is more advantegous than the other method, Method withLoop is more general and flexible, as it can handle arrays of any size and contents. But in terms of time efficiency, there is no difference between them.

Resit Aydın 2001040040 1/9

Subject : Date : ...../...../.....

(4) No, we can't solve this problem in constant time, because if the array is not sorted, we have to iterate over whole array to find the specific integer, even if the array is sorted, we could find it in $log(n)$ time by using Binary-search method. So we can't find this problem in constant time.

(5) In my algorithm, to avoid using quadratic time algorithms, which are very inefficient, I first found minimum and maximum values of both A and B arrays. The reason I found not only minimum values of them but also maximum of them is because if they are both negative values, if we find the product of them, we end up with finding their max value of product. To sum up, with this algorithm, I compared 4 possible combination of the product of minimum and maximum values and found the minimum of them then returned that value.

Pseudo code of the algorithm:

```
function minProduct(A, B):

    min1=10000
    min2=10000
    max1= -10000
    max2=-10000

    for i in range(len(A)):
        if(A[i]<min1) min1 = A[i]

        if(A[i]>max1) max1=A[i]
```

→

Resit Aydın 2001040019

Subject :

Date :...../...../.....

```
for i in range(len(B)):
if(B[i]<min2) min2=B[i]
if(B[i]>max2) max2=B[i]

if(min1*min2<max1*max2 and min1*min2<min1*max2
    and min1*min2<max1*min2) return min1*min2
elif(max1*max2<min1*min2 and max1*max2<min1*max2 and
    max1*max2<max1*min2) return max1*max2
elif(min1*max2<min1*min2 and min1*max2<max1*max2 and
    min1*max2<max1*min2) return min1*max2
return max1*min2
```

Time complexity of the algorithm at the worst case:

$T(n,m) = O(n+m)$