Assignment-1

Name of the student : Challa. Ravi teja

Register Number of Student : 192325065

Course code : CSA0670

Name of the Subject : Design and Analysis of Algorithm.

Date of Submission : 20/06/2024

1. Solve the following recurrence relation.

a) $x(n) = x(n-1) + 5$ for $n > 1$ with $x(1) = 0$

A: Step 1: write down the first two terms to identify the pattern.

$x(1) = 0$

$x(2) = x(1) + 5 = 5$

$x(3) = x(2) + 5 = 10$

$x(4) = x(3) + 5 = 15$

Step 2: Identify the pattern (or) the general term.

→ The first term $x(1) = 0$

→ The common difference $d = 5$

The general formula for the n^{th} term of an AP is

$$x(n) = x(1) + (x^{n}-1) \cdot d$$

Substituting the given values

$$x(n) = 0 + (n-1) \cdot 5 = 5(n-1)$$

The Solution is $x(n) = 5(n-1)$.

b) $x(n) = 3x(n-1)$ for $n > 1$ with $x(1) = 4$

A: Step 1: write down the first two terms to identify the pattern.

$x(1) = 4$

$x(2) = 3x(1) = 3 \cdot 4 = 12$

$x(3) = 3x(2) = 24$

$x(4) = 3x(3) = 36$

Step 2: Identify the general term

→ The first term $x(1) = 4$

→ The common ratio $r = 3$

The general formula for the n^{th} term of a GP is

$$x(n) = x(1) \cdot r^{n-1}$$

Substituting the given values. $x(n) = 4 \cdot 3^{n-1}$

The Solution is $x(n) = 4 \cdot 3^{n-1}$

c) $x(n) = x(n/2)+n$ for $n>1$ with $x(1) = 1$ (Solve for $n=2^k$)

A: For $n=2^k$, we can write recurrence interms of k.

1. Substitute $n=2^k$ in the reccurence.

$$x(2^k) = x(2^{k-1})+2^k$$

2. Write down the first few terms to identify the pattern

$$x(1)=1$$
$$x(2)=x(2^1)= x(1)+2 = 3$$
$$x(4)= x(2^2)= x(2)+4 = 3+4=7$$
$$x(8)= x(2^3) = x(4)+8 = 7+8 = 15$$

3. Identify the general term by finding the pattern we observe that. $x(2^k)= x(2^{k-1})+2^k$

we sum the series. $x(2^k) = 2^k+2^{k-1}+2^{k-2}+ \text{----}$

Since $x(1) = 1$

$x(2^k) = 2^k+2^{k-1}+2^{k-2}+ \text{----}$

The geometric series with the term $a=2$ and the last term 2^k except for the additional $+1$ term. The sum of a geometric series S with ratio $r=2$ is given by. $S = a\frac{r^n-1}{r-1}$

Here $a=2$, $r=2$ and $n=k$.

$$S = \frac{2^{2k}-1}{2-1} = 2(2^k-1) = 2^{k+1}$$

Adding the $+1$ term

$$x(2^k) = 2^{k+1}-2+1 = 2^{k+1}-1$$

Solution is $x(2^k) = 2^{k+1}-1$.

d) $x(n) = x(n/3)+1$ for $n>1$ with $x(1) = 1$ (Solve for $n=3^k$)

A: For $n=3^k$, we can write the reccurence interms of the k

1. Substitute $n=3^k$ in the reccurence $x(3^k) = x(3^{k-1})+1$

2. Write down the first few terms to identify the pattern

$x(1)=1 \Rightarrow x(27) = x(3^3) = x(9)+1 = 4$.

3. Identify the general term:

we observe that $x(3^k) = x(3^{k-1})+1$

The Solution is $x(3^k) = k+1$

2. Evaluate the following reecurence complexity

(i) $T(n) = T(n/2) + 1$, where $n = 2^k$ for all $k \geq 0$.

A: The recurrence relation can be solved using iteration method

1. Substitute $n = 2^k$ in the recurrence.

2. Iterate the recurrence

for $k=0$: $T(2^0) = T(1) = T(1)$

$k=1$: $T(2^1) = T(1) + 1$

$k=2$: $T(2^2) = T(8) = T(n)+1 = T(1)+2+1 = T(1)+2$

$k=3$: $T(2^3) = T(8) = T(n)+1 = T(1)+2+1 = T(1)+3$

3. generalize the pattern

$T(2^k) = T(1) + k$.

Since $n = 2^k$, $k = \log_2 n$.

$T(n) = T(2^k) = T(1) + \log_2 n$

4. Assume $T(1)$ is a constant C

$T(n) = C + \log_2 n$

The solution is $T(n) = O(\log 0)$

(ii) $T(n) = T(n/3) + T(2n/3) + Cn$. where C is constant and n. is input size.

A: The recurence can be solved using the master's theorem for divide and conquer recurence of the form.

$T(n) = at(n/b) + f(n)$

where $a=2, b=3$ and $f(n) = (n)$

lets determine the value of $\log_b a$

$\log_b a = \log_3 2$

using the properties of algorithm.

$\log_3 2 = \frac{\log 2}{\log 3}$.

Now we compare $F(n) = Cn$ with $n\log_3 2$

$F(n) = O(n)$

$n = n^1$

Since $\log_3^2$ we are in the third case of the master's Theorem.

$F(n) = O(n^c)$ with $c > \log_b^a$

The solution is: $T(n) = O(F(n)) = O(cn) = O(n)$

3. Consider the following reccurence algorithm?

```
min [A(0.---- n-2)]
if n=1 return A[0]
else temp: min (A[0----n-2])
    if temp <= A(n-1) return temp
    else return A(n-1]
```

a) what does this algorithm compute?

The given algorithm, min[A(0,--- n-2]) compute the minimum value in the array 'A' from index '0' for 'n-1' it does this by recursively finding the minimum value in the Sub array. A[0.---n-2] and then comparing it with the last element A[n-1] to determine the overall maximum value.

b) Setup a recurrence relation for the algorithm basic operation count and solve it.

The Solution is

$T(n) = n$

This means the algorithm performs n basic operations for an input array of size n.

∴ The Solution is

$T(n) = n$ //

4. Analyze the order of growth.

(i) $F(n) = 2n^2+5$ and $g(n) = 7n$ use the $\Omega\, g(n)$ notation.

To analyze the order of growth and use the Ω notation, we need to compare the given function $f(n)$ and $g(n)$

given functions:

$$F(n) = 2n^2+5$$
$$g(n) = 7n$$

Order of growth using $\Omega(g(n))$ notation.

The Notation $\Omega(g(n))$ describe a lower bound on the growth rate that for sufficiently large n, $f(n)$, grows at least as far as $g(n)$

$$F(n) = C \cdot g(n)$$

less analyze $F(n) = 2n^2+5$ with respect to $g(n) = 7n$.

1. Identify dominant terms:

→ The dominant term in $F(n)$ is $2n^2$ since it grows faster then the constant term as n increases.

→ The dominant term in $g(n)$ is $7n$.

2. Establish the inequality.

→ Ignore the lower order term 5 for larger.

$$2n^2 \geq 7Cn$$

→ Divide both sides by n.

$$2n \geq 7C$$

→ Solve for n:

$$n \geq 7C/2$$

4. choose constants.

let $C=1$

$$n \geq \frac{7 \cdot 1}{2} = 3.5$$

∴ For $n \geq n$, The inequality holds.

$2n^2+5 \geq 7n$. For all $n \geq n$

we have shown that there exist constant $c=1$ and $n_0 = n$

Such that for all $n \geq n_0$

$$2n^2+5 \geq 7n$$

Thus, we can conclude that:

$$F(n) = 2n^2+5 = \Omega(7n)$$

In Ω rotation, The domainant term $2n^2$ in $F(n)$ clearly grows faster than $7n$. Hence

$$F(n) = \Omega(n^2)$$

However, for the specific comparision asked $F(n) = \Omega(7n)$ is also correct.

Showing that $F(n)$ grows at least as Fast as $7n$.