Instituto Politécnico Nacional

ESCOM

Javier Fuentes Mora

Ejercicios

7° Funcion Recursiva

$2 + T(n-1) + T(n-2) + T(n-3)$

$T(n) = 2 + T(n-1) + T(n-2) + T(n-3)$

$T(n) - T(n-1) - T(n-2) - T(n-3) = 2$ $\qquad b^n p(n)$

$2(1)$

$x^k = T(n) \qquad d = 0$

$x^3 = T(n)$

$(x^3 - x^2 - x^1 - 1)(x - 2) = 0$

$T(0) = 1$

$T(1) = 1$

$T(2) =$

$r_1 = 2 \qquad r_3 \approx -0.41964 - 0.60629i$

$r_2 = 1.8365 \qquad r_4 \approx -0.41964 + 0.60629i$

$T(n) = C_1(2)^n + C_2(1.8365)^n$

$T(0) = 1 = C_1(2)^0 + C_2(1.8365)^0 = C_1 + C_2$

$T(1) = 1 = C_1(2) + C_2(1.8365)$

$\underline{T(n) = -5.1162(2)^n + 6.11(1.83)^n}$

$\in \Theta(a^n)$

$C_1 = 1 - C_2 \qquad 2C_1 = 1 - C_2(1.8365)$

$2(1 - C_2) = 1 - C_2(1.8365)$

$2 - 2C_2 = 1 - C_2(1.8365)$

$2 - 1 = C_2(0.1635)$

$1 = C_2(0.1635)$

$C_2 = \dfrac{1}{0.1635} \approx 6.1162$

$C_1 = 1 - C_2$

$C_1 = -5.1162$

(2)

```
int Triboracci (int num)
{
    if (num == 0)
        return 0;
    else if (num == 1 || num == 2)
        return 1;
    else
        return Triboracci (num-1) + Triboracci(num-2) + Triboracci(num-3);
}
```

$T(0) = 1$

$T(1) = 2$

$T(2) = 2$

$\underline{4 + T(n-1) + T(n-2) + T(n-3)}$

$T(n) - T(n-1) - T(n-2) - T(n-3) = 4$

$x^k = T(n)$ $\quad 4^n\ (\rho(n))$

$x^3 = T(n)$ $\quad 4^1$

$b = 4 \quad d = 0$

$(x^3 - x^2 - x^1 - 1)(x - 1)$

$r_1 = 1$

$r_2 \approx 1.93929$

$T(n) = C_1 (1)^n + C_2 (1.839)^n$

$\in \Theta(a^n)$

3. Resolver las siguientes ecuaciones y dar su orden de complejidad

- $T(n) = 3T(n-1) + 4T(n-2) \Longrightarrow n > 1 \quad T(0) = 0 ; \; T(1) = 1.$

Homogenea

$$T(n) - 3T(n-1) - 4T(n-2) = 0$$

$$x^2 = T(n) \qquad r_1 = 4$$

$$r_2 = -1$$

$$x^2 - 3x - 4 = 0$$

$$\left(x - \frac{3}{2}\right)^2 = 4 + \left(\frac{3}{2}\right)^2$$

$$\left(x - \frac{3}{2}\right)^2 = 4 + \frac{9}{4}$$

$$\left(x - \frac{3}{2}\right)^2 = \frac{25}{4}$$

$$x - \frac{3}{2} = \pm \frac{5}{2}$$

$$x_1 = +\frac{5}{2} + \frac{3}{2} = \frac{8}{2} = 4$$

$$x_2 = -\frac{5}{2} + \frac{3}{2} = \frac{-2}{2} = -1$$

$$T(n) = C_1(-1)^n + C_2(4)^n$$

$$T(0) = 0 = C_1 + C_2 \qquad C_1 = -C_2$$

$$T(1) = 1 = C_1(-1)^1 + C_2(4)^1$$

$$-C_1 + 4C_2 = 1$$

$$C_1 = 4C_2 - 1$$

$$4C_2 - 1 = -C_2$$

$$5C_2 = 1$$

$$C_2 = \frac{1}{5}$$

$$C_1 = \frac{-1}{5}$$

$$T(n) = -\frac{1}{5}(-1)^n + \frac{1}{5}(4)^n \quad \in O(a^n).$$

3.2

$T(n) = 3T(n-1) + 4T(n-2) + (n+5)\,2^n \Rightarrow n > 1;\; T(0)=5,\; T(1)=27$

$T(n) - 3T(n-1) - 4T(n-2) = 2^n(n+5)$ $\quad T(0)=5$

$\quad T(1)=27$

$x^k = x^2 = T(n) \quad b^n\, p(n) \quad 2^n(p(n)) \quad d=1$

$(x^2 - 3x - 4)(x-2)^2 = 0$

$x^2 - 3x - 4 = 0$

$\left(x - \frac{3}{2}\right)^2 = 4 + \left(\frac{3}{2}\right)^2$

$\left(x - \frac{3}{2}\right)^2 = 4 + \frac{9}{4}$

$x_1 = 4$

$x_2 = -1$

$(x-2)^2 = 0$

$x = +2$

$r_1 = 2$

$r_2 = 2$

$T(n) = C_1(2)^n + C_2(4)^n + C_3(-1)^n$

$T(0) = 5 = C_1 + C_2 + C_3$

$C_1 = 5 - C_2 - C_3 \to 1$

$2 \to C_1 + C_2 + C_3 = 5$

$T(1) = 27 = C_1(-1)^1 + C_2(4) + C_3(2)^1 + C_4(2)$

$T(2) = 3T(1) + 4T(0) + (2+5)2^2 = 3(27) + 4(5) + 28 = 129 = (-1)^2C_1 + 4^2C_2 + 2^2C_3 + 2^2C_4(2)$

$= C_1 + 16C_2 + 4C_3 + 8C_4$

$T(3) = 3T(2) + 4T(1) + (3+5)2^3 = 3(129) + 4(27) + 64 = 559 = (-1)^3C_1 + 4^3C_2 + 2^3C_3 + 2^3C_4(3)$

$= -C_1 + 64C_2 + 8C_3 + 24C_4$

$C_1 = \frac{13}{45}, \quad C_2 = \frac{48}{5}; \quad C_3 = -\frac{44}{9}; \quad C_4 = -\frac{2}{3}$

$\therefore T(n) = (-1)^n\frac{13}{45} + 4^n\frac{48}{5} - 2^n\frac{44}{9} - 2^n\frac{2}{3}n \in O(a^n)$

3.3 $T(n) - 2T(n-1) = 3^n$ $\rightarrow n \geq 2;\ T(0)=0$

$T(1)=1$

$x^k = T(n)$ $b=3$

x^1 $d=0$

$(x-2)(x-3)^1 = 0$

$r_1 = 2$

$r_2 = 3$

$T(n) = C_1(2)^n + C_2(3)^n = 0$

$T(0) = 0 = C_1 + C_2 \rightarrow C_1 = -C_2$

$T(1) = 1 = 2C_1 + 3C_2$

$1 = 2(-C_2) + 3C_2$

$C_2 = 1$ $\therefore C_1 = -1$

Sustituyendo

$T(n) = -2^n + 3^n$ $\in O(a^n)$

4º Calcular la cota de complejidad del algoritmo de busqueda binaria recursiva

$$T(n) = 3 + T(n/2)$$

$$T(0) = O(c)$$

$$a = 1 \quad b = 2 \qquad f(n) = 3 \in O(1)$$

Probando 1

$$O(1) = O(n^{\log_2(1)} - 1) \neq O(n^{-1})$$

Probando 2

$$O(1) = O(n^{\log_2 1}) = O(n^{0})$$

$\therefore$

$$T(n) = \Theta(n^{\log_2 1} \log(n)) = \underline{O(\log n)}$$

50 Calcular Merge - Sort

$$T(n) = 2T\left(\frac{n}{2}\right) + n$$

$a = 2 \quad b = 2 \quad f(n) = n$

$\varepsilon = 1$ X

Probando 1 $O(n^{\cancel{\log_2(2)}^{1} - 1}) = O(n^{0}) = O(n)$ /

Probando 2 $O(n^{\cancel{\log_2(2)}^{1}}) = O(n^1) = O(n)$ //

$\therefore T(n) = \Theta(n^{\cancel{\log_2 2}^{1}} \log(n)) = \Theta(n \log n)$

6 Calcular la cota de complejidad que tendrán los algoritmos con los sig. modelos recurrentes

$T(n) = 3T\left(\frac{n}{3}\right) + 4T\left(\frac{n}{2}\right) + 2n^2 + n$

$T_1(n) = 3T\left(\frac{n}{3}\right) + 2n^2$

$T_2(n) = 4T\left(\frac{n}{2}\right) + n$

$T_n = T_1(n) + T_2(n)$

$T_2(n) = 4T\left(\frac{n}{2}\right) + n$

$a = 4 \quad b = 2 \quad f(n) = n$

Probando 1 con $\varepsilon = 1$

$O(n^{\log_2(4) - 1}) = O(n) = f(n)$

$\therefore T_2(n) = \Theta(n^2)$

$T_1(n) = 3T\left(\frac{n}{3}\right) + 2n^2$

$a = 3 \quad b = 3 \quad f(n) = n^2$

Probando 1

$n^2 = O(n^{\log_3 3 - 1})$

$n^2 \neq O(n^0)$

Probando 3

$\Omega(n^{\cancel{\log_3 3}^{1} + \varepsilon})$ con $\varepsilon = 1$

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

$T_1(n) = \Theta(n^2)$ /

$T(n) = \Theta(n^2) + \Theta(n^2) \in \Theta n^2$ /

6.2

$T(n) = T(n-1) + T(n-2) + T(n/2)$ si $n > 1$

$T(0) = 5 \quad T(1) = 1$

$T_1(n) = T(n-1) + T(n-2)$

$T_2(2) = T(n/2)$

Para $T_1(n)$

$T_1(n) - T(n-1) - T(n-2) = 0$

$x^k = T_1(n)$

$x^2 = T_1(n)$

$(x^2 - x - 1) = 0$

$\left(x^2 - \frac{1}{2}\right)^2 = 1 + \left(\frac{1}{2}\right)^2$

$\left(x - \frac{1}{2}\right)^2 = 1 + \frac{1}{4} = \frac{5}{4}$

$r_1 = \frac{1 + \sqrt{5}}{2}$

$r_2 = \frac{1 - \sqrt{5}}{2}$

$T(n) = C_1\left(\frac{1+\sqrt{5}}{2}\right)^n + C_2\left(\frac{1-\sqrt{5}}{2}\right)^n$

$T(0) = 5 = C_1 + C_2 \qquad C_1 = 5 - C_2$

$T(1) = 1 = C_1\left(\frac{1+\sqrt{5}}{2}\right) + C_2\left(\frac{1-\sqrt{5}}{2}\right)$

$1 = (5 - C_2)\left(\frac{1+\sqrt{5}}{2}\right) + \frac{C_2 - C_2\sqrt{5}}{2}$

$1 = \frac{5}{2} + \frac{5\sqrt{5}}{2} - \frac{C_2}{2} - \cancel{\frac{C_2\sqrt{5}}{2}} + \cancel{\frac{C_2}{2}} - \frac{C_2\sqrt{5}}{2}$

$1 = -C_2\sqrt{5} + \frac{C_2}{2} + \frac{5 + 5\sqrt{5}}{2}$

$-7.09 = -C_2\sqrt{5} + \frac{C_2}{2}$

$-7.09 = -1.2360\, C_2$

$C_2 = 5.7362$

$C_1 = -0.7362$

$T_1(n) = (-0.7362)\left(\frac{1+\sqrt{5}}{2}\right)^n + 5.73\left(\frac{1-\sqrt{5}}{2}\right)^n$

$\in \Theta\, a^{(n)}$

$T_2(n) = T\left(\frac{n}{2}\right)$

$a = 1 \quad b = 2 \quad F(n) = 0$

$\Theta\ \{ F(n) = \Theta(1)$

$③\ O(1) = \Theta(n^{\log_2 a}) = O(1)$

$T_2(n)\ (n^0 \log n) = \Theta(\log n)$

$T(n) = T_1(n) + T_2(n)$

$T(n) = \Theta a^n + \Theta \log n = \underline{\Theta(a^n)}$

6.3

$$T(n) = T\left(\frac{n}{2}\right) + 2T\left(\frac{n}{4}\right) + 2 = 0$$

$$T(n) = T\left(\frac{n}{2}\right) + 2T\left(\frac{n}{4}\right) + 2$$

$$T_1(n) = T\left(\frac{n}{2}\right)$$

$$T_2(n) = +2T\left(\frac{n}{4}\right) + 2$$

$$T_1(n) = T\left(\frac{n}{2}\right)$$

$$T_1(n) = \Theta(\log n)$$

$$T_2(n) = +2T\left(\frac{n}{4}\right) + 2$$

$a = +2$

$b = 4$

?

$f(n) = +2 \in O(1)$

$\Theta(1) = O(n^{\log_4(+2) - 1/2})$

$\Theta(1) = O(n^0)$

$\therefore T_2(n) = \Theta(n^{1/2})$

$$T(n) = T_1(n) + T_2(n)$$

$$= O(\log n) + \Theta(n^{1/2})$$

$$T(n) = \Theta(n^{1/2})$$

Conclusión

Mediante esta práctica pude comprender más a fondo el análisis recursivo me permitió comprender mejor la forma en que se ejecutan los programas y ya que al principio se dificulto por lo ejercicios que tenían raíces complejas , y los ejercicios que combinaban el teorema maestro con los otros métodos , me ayudo a saber diferenciar y combinar las soluciones de ambos.