DAA Упражнение 5 (Рекурентни уравнения)

Чрез рекурентни уравнения ще разглеждаме сложността на рекурсивни алгоритми. Всички рекурентни уравнения, които ще разглеждаме са със строго намаляващ аргумент. В курса не ни интересува точно решение на рекурентното уравнение, а само търсим асимптотиката затова ще изпускаме базата на рекурентните уравнения и ще приемаме, че тя е константа за достатъчно малки стойности. Има няколко начина за решаване на рекурентни уравнения:

I) Характеристично уравнение: То е от вида $T(n) = \overbrace{a_1 . T(n-b_1) + \dots + a_k T(n-b_k)}^{\text{хомогенна част}} + \underbrace{c_1^n . P_1(n) + \dots + c_m^n P_m(n)}_{\text{нехомогенна част}}$, където a_i, b_j, c_k са константи $\forall i \forall j \forall k$, b-тата са две по две различни и цели, c-тата са различни помежду си, $P(n)$ са полиноми с неотрицателна цяла степен.

зад.1/ Каква е асимптотиката на $T(n) = 4T(n-2) + n.2^n + 4.3^n$

Решение: От хомогенната част имаме $x^n = 4x^{n-2} \leftrightarrow x^2 = 4 \Rightarrow \{-2, 2\}_M$ мултимножество

От нехомогенната част имаме: добавяме толкова на брой 2ки, колкото е степента на полинома + 1

$$\left.\begin{array}{l} -\ n2^n \Rightarrow \{2,2\}_M \\ -\ 4.3^n \Rightarrow \{3\}_M \end{array}\right\} \{2,2,3\}$$

След обединяване на двете мултимножества получаваме $\{-2,2,2,2,3\}_M$

Получаваме, че $T(n) = c_1.(-2)^n + c_2.2^n + c_3.n2^n + c_4.n^2.2^n + c_5.3^n = \boxed{\Theta(3^n)}$ за някакви константи $c_1, \dots, c_5 > 0$. В решенията ще изпускаме последното изречение, защото можем да забележим че асимптотиката винаги се определя от най-голямата константа в мулти множеството и броят на срещанията ѝ.

зад.2/ Каква е асимптотиката на $T(n) = 3T(n-1) - 2T(n-2) +$ [redacted] $+ 2^n.n^3 + \underset{1^n.n}{n} + 2^n.10n^2 + \underset{1^n.5}{5} + \underset{3^n.n^0}{3^n}$

Решение: $T(n) = 3T(n-1) - 2T(n-2) + 2^n(n^3 + 10n^2) + 1^n(n+5) + 3^n.n^0$

От хомогенната част имаме: $x^n = 3x^{n-1} + 2.x^{n-2} \leftrightarrow x^2 - 3x + 2 = 0 \Rightarrow \{1,2\}_M$

Забележка: Припомнете си формулите на Виет и метода на Хорнер!

От нехомогенната част имаме $\{1,1,2,2,2,2,3\}_M$. След обединяването на двете мултимножества получаваме $\{1,1,1,2,2,2,2,2,3\}_M \Rightarrow T(n) = \boxed{\Theta(3^n)}$

зад. 3/ $T(n) = T(n-2) + T(n-4) + \dots + \underbrace{T(n\%2)}_{T(0) \text{ или } T(1) \text{ спрямо четността на } n}$

Решение: Ще решим това рекурентно уравнение със следният трик. Когато имаме пълна история на рекурентно уравнение вадим две съседни. Тоест $T(n) - T(n-2) = (T(n-2) + T(n-4) + \dots + T(0)) - (T(n-4) + \dots + T(0)) = T(n-2)$

Откъдето $T(n) - T(n-2) = T(n-2)$

$$T(n) = 2T(n-2)$$

$x^n = 2 \cdot x^{n-2} \leftrightarrow x^2 = 2 \Rightarrow \{-\sqrt{2}, \sqrt{2}\}_M \Rightarrow T(n) = \boxed{\Theta(\sqrt{2}^n)}$

Итератори: $T(n) = T(\_) + \dots$ $\begin{cases} n \to n-c \ (\approx \frac{n}{c} \text{ стъпки да се стигне до } 0) = \Theta(n) & c>0 \\ n \to \frac{n}{c} \ (\approx \log_c n \text{ стъпки да се стигне до } 0) = \Theta(\lg n) & c>1 \\ n \to \sqrt[c]{n} = \Theta(\lg \lg n) \end{cases}$

Следствие: $T(n) = 2T(\_) + \dots$ $\begin{cases} n \to n-c = \Theta(2^n) \\ n \to \frac{n}{c} = \Theta(n) \\ n \to \sqrt[c]{n} = \Theta(\lg n) \end{cases}$

II) Развиване (неформално, приложимо само ако рек. уравнение е от 1ред)

зад. 4/ $T(n) = T(n-1) + \lg(n) =$

$= T(n-2) + \lg(n-1) + \lg(n) =$

$= T(n-3) + \lg(n-2) + \lg(n-1) + \lg(n) = \dots$ неформално

$\dots = T(1) + \lg(2) + \dots + \lg(n) = \Theta(1) + \lg(2.3,\dots,n) = \lg(n!) + \Theta(1)$

$= \Theta(n \lg n)$ от следствието на А. на С.

Проблемът с развиването е, че е неформален метод за решаване и полученото трябва да го докажем с индукция, което ще видим как става по-късно.

зад. 5/ $T(n) = 2T(\frac{n}{2}) + 1 =$

$= 2(2T(\frac{n}{4}) + 1) + 1 = 4T(\frac{n}{4}) + 2 + 1 =$

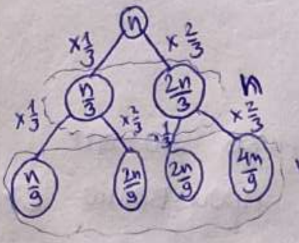
$= 4(2T(\frac{n}{8}) + 1) + 2 + 1 = 8T(\frac{n}{8}) + 4 + 2 + 1 = \dots$

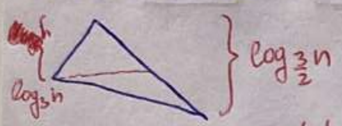
$\dots = n \cdot T(1) + \frac{n}{2} + \frac{n}{4} + \dots + 1 = n \cdot \Theta(1) + \Theta(n) = \Theta(n)$

III Дърво на рекурсията (неформално)

зад.6 / $T(n) = T(\frac{n}{3}) + T(\frac{2n}{3}) + n$

Нека да си представим, че това е някаква рекурсивна програма, която има 2 рекурсивни извиквания и извършва n на брой операции във всяко извикване. Понеже в някакъв момент ще достигнем базата и на всяко ниво ние извършваме общо n на брой стъпки, то трябва да видим каква е височината на дървото на рекурсията (за да сметнем височина * брой стъпки на ниво). В най-добрият случай височината на дървото е $\log_3 n$ и в най-лошия случай е $\log_{\frac{3}{2}} n$. Следователно предполагаме, че асимптотиката е $n\lg n$, но това трябва да се докаже чрез индукция!





IV Чрез полагане (формално, ако положеното се реши чрез ХУ, МТ или индукция)

зад.7 / $T(n) = 2T(\sqrt{n}) + \lg n$

Полагаме $n = 2^{2^m}$. Тогава $T(2^{2^m}) = 2T(2^{2^{\frac{m}{2}}}) + \lg 2^{2^m}$

$T(2^{2^m}) = 2T(2^{2^{m-1}}) + 2^m$ Нека пол. $S(m) = T(2^{2^m})$

$S(m) = 2S(m-1) + 2^m$

От хомогенната част имаме $\{2\}_M$ и от нехомогенната част имаме $\{2\}_M$. Следователно $S(m) = \Theta(m.2^m)$. Обаче ние искаме решението за $T(n)$. Тогава връщаме полагането и получаваме $T(n) = \Theta(\lg\lg n . 2^{\lg\lg n}) =$
$= \boxed{\Theta(\lg\lg n . \lg n)}$

V Чрез индукция

зад.8 / $T(n) = 2T(n-1) + n$ Да се докаже, че $T(n) = \Theta(2^n)$

Решение: Допускаме, че $\forall m < n\ \exists c > 0\ \exists n_0 > 0\ \forall m \ge n_0\ T(m) \le c.2^m$

Ще докажем за n. $T(n) = 2T(n-1) + n \overset{\text{от}}{\underset{\text{И.П.}}{\le}} 2c(2^{n-1}) + n \overset{?}{\le} c.2^n$ ← питаме се дали това е изпълнено

Тоест $c.2^n + n \le c.2^n$, но това не е изпълнено за никое $n \in \mathbb{R}^+$ и $c \in \mathbb{R}^+$

Проблемът е там, че или сложността на рекурентното уравнение не е $\Theta(2^n)$, или не сме засилили достатъчно твърдението.

Нека се опитаме да засилим твърдението, като извадим, следното

2 опит) Допускаме, че $\forall m < n\ \exists c > 0\ \exists b > 0\ \exists n_0 \in \mathbb{N}\ \forall m \geq n_0 : T(m) \leq c.2^m - b.m$

Ще докажем за n. $T(n) = 2T(n-1) + n \underset{\text{и.п.}}{\overset{\text{от}}{\leq}} 2(c.2^{n-1} - b(n-1)) + n \overset{?}{\leq} c.2^n - bn$

Тоест се питаме дали $\cancel{c.2^n} - 2bn + 2b + n \overset{?}{\leq} \cancel{c.2^n} - bn$

$$-bn + n + 2b \overset{?}{\leq} 0$$
$$n(1-b) + 2b \overset{?}{\leq} 0$$

Ако $(1-b).n$ е отрицателно, то със сигурност от някакъв момент нататък ще доминира $2b$ и ще е ≤ 0. Тоест за произволно $c > 0$ $b > 1$ е изпълнено, че $T(n) = O(2^n - bn)$. Нека $\boxed{c=1}$ и $\boxed{b=2}$ и $n_0 = 10$.

Остана да докажем, че $T(n) = \Omega(2^n)$

Допускаме, че $\forall m < n\ \exists d > 0\ \exists n_0 \in \mathbb{N}\ \forall m \geq n_0 : T(m) \geq d.2^m$

Ще докажем за n. $T(n) = 2T(n-1) + n \underset{\text{и.п.}}{\overset{\text{от}}{\geq}} 2(d.2^{n-1}) + n \overset{?}{\geq} d.2^n$

Тоест се питаме дали $\cancel{d.2^n} + n \geq \cancel{d.2^n}$, което е изпълнено за $\forall d > 0$ и $n_0 \in \mathbb{N}$.

Нека $d = 1$ и $n_0 = 1$. ∎

Заб: При засилване на И.П. (ако е необходимо), то се случва само в едната посока (тоест ако засилим O посоката, то Ω няма нужда да се засилва). Освен това по време на контролно или изпит, ако има да се решава рекурентно уравнение чрез индукция, то си спестете време, като засилите максимално твърдението (тоест ако трябва да докажете, че нещо е $\theta(n^3)$ засилете O посоката като извадите $b.n^2$).