# Асимптотика на рек. уравнения

**Заб** Рекурентните уравнения, които ще разглеждаме ще са със строго намаляващ аргумент. Също така ще считаме, че $\exists\, n_0\ \forall\, n < n_0 : T(n)$ се изпълнява за константно време. За леснота, ще си нагаждаме $n_0$ с 0/1/2.. някоя малка стойност, която да ни е удобна за работа над конкретната задача.

**Зад. 1** //Използвайте: характеристично уравнение

$T(n) = 4\,T(n-2) + n2^n + 4.3^n$

От хомогенната част имаме: $x^2 = 4\,x^0 \mapsto x_{1,2} = \pm 2 \mapsto \{2,\ -2\}_M$

От нехомогенната част имаме: $n2^n \mapsto \{2,\ 2\}_M$ & $4\,n^0\,3^n \mapsto \{3\}_M$ или общо $\{2,\ 2,\ 3\}_M$

Общо имаме: $\{-2,\ 2,\ 2,\ 2,\ 3\}_M$

Оттук получаваме $T(n) = c_1(-2)^n + c_2\,2^n + c_3\,n2^n + c_4\,n^2\,2^n + c_5\,3^n \overset{[1]}{=} \theta(3^n)$.

**[1]** Разбираме се, че в рамките на курса коефицента пред най-значимото асимптотично събираемо е положително.

**Зад. 2** //Използвайте: характеристично уравнение

$T(n) = 2\,T(n-1) - T(n-2)$

Тук имаме само хомогенна част: $x^2 = 2\,x - 1 \mapsto \{1,\ 1\}_M$

Оттук получаваме $T(n) = c_1\,1^n + c_2\,n1^n = \theta(n)$.

**Зад. 3** //Използвайте: развиване

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

$$T(n) = T(n-1) + n = (\underbrace{T(n-2) + (n-1)}_{T(n-1)}) + n = (\underbrace{T(n-3) + (n-2)}_{T(n-2)}) + (n-1) + n = \ldots = \underbrace{T(0)}_{\text{const}} + 1 + 2 + \ldots + (n-1) + n = \theta\left(n^2\right).$$

Ако не е написано експлиситно точкуването на дадената задача и напишем само това, то *по подразбиране* би носело точно 0 точки на контролно и/или домашно. Трябва да го докажем формално, че е $\theta\left(n^2\right)$. Това се прави с индукция. Можем да докажем директно за $\theta\left(n^2\right)$, а може и $O\left(n^2\right)$ & $\Omega\left(n^2\right)$, откъдето ще следва $\theta\left(n^2\right)$. Понякога "омега посоката" би ни била безплатна. Нека сега го докажем за $O\left(n^2\right)$ & $\Omega\left(n^2\right)$ за да се видят приликите и разликите.

Нека $n_{\text{st}} \in \mathbb{N}_0$ е достатъчно голямо число. [2]

***a) $T(n) = O\left(n^2\right)$***

Ще докажем, че $T(n) = O\left(n^2\right)$.

По деф. на $O\left(n^2\right) = \left\{f \mid \exists\, c > 0\ \exists\, n_0 \in \mathbb{N}_0\ \forall\, n \geq n_0 \text{ е изпълнено } 0 \leq f(n) \leq c.n^2\right\}$.

Тоест търсим константа $c > 0$ и търсим константа $n_0 \in \mathbb{N}_0$ такива, че ни е изпълнено $\forall\, n \geq n_0 : 0 \leq T(n) \leq c.n^2$. Приемаме неравенството $T(n) \geq 0$ за тривиално изпълнено (в рамките на курса, го ползваме наготово.. все пак няма как една ф-я да върши отрицателна работа).

Нека $b = \max\left\{\frac{T(1)}{1^2},\ \frac{T(2)}{2^2},\ \frac{T(3)}{3^2},\ \ldots,\ \frac{T(n_{\text{st}}-1)}{(n_{\text{st}}-1)^2},\ ?\right\}$. [3]

Ще докажем, че за $c = b$ и за някое $n_0$ (ще установим по-надолу), че $\forall\, n \geq n_0 : T(n) \leq b.n^2$.

**База:**

Нека $k \in \{1, \ldots, n_{st} - 1\}$. Ще докажем, че $T(k) \leq b.k^2$.

От деф. на $b = \max\left\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, ?\right\} \overset{1 \leq k \leq n_{st}-1}{\Rightarrow} b \geq \frac{T(k)}{k^2} \Rightarrow T(k) \leq b.k^2$, което искахме да докажем.

**ИП:**

Нека допуснем, че е изпълнено $\forall\, m < n : T(m) \leq b.m^2$.

**ИС:**

Ще докажем, че е изпълнено за $n$, т.е $T(n) \leq b.n^2$.

$T(n) \overset{\text{def}}{=} T(n-1) + n \overset{\text{ИП}}{\leq} b(n-1)^2 + n = bn^2 - 2\,bn + b + n \overset{?}{\leq} bn^2$

$-2\,bn + b + n \overset{?}{\leq} 0$

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

Може да забележим, че при $1 - 2\,b \leq -b$, горното неравенство ще е изпълнено за всяко $n \geq 1$. Оттук си избираме $n_0 \overset{(*)}{=} 1$, което не бяхме сигурни по-горе.

Остана да видим $1 - 2\,b \leq -b \leftrightarrow b \geq 1$. Това не сме си го подсигурили. Затова горе, където дефинирахме $b = \max\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, ?\}$, този въпросителен знак се въщаме и го заместваме с 1.

*Тъй като това е статичен .pdf, то няма как да се върна и да го заместя горе, но на контролно/домашно просто си оставете празно място и го попълнете, когато го откриете. Нарочно не го попълвам, защото би изглеждало все едно е паднало от небето тази единица. За по-опитен в тази сфера, ще знае, че е някоя константа, която ни е нужна по-надолу в доказателството.*

Тоест в крайна сметка имаме $b = \max\left\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, 1\right\}$.

И съответно показахме, че за $c = b$ & $n_0 = 1$, то $\forall\, n \geq \underbrace{n_0}_{1} : T(n) \leq c.n^2$. Оттук доказахме, че $T(n) = O\left(n^2\right)$.

**b) $T(n) = \Omega\left(n^2\right)$**

Ще докажем, че $T(n) = \Omega\left(n^2\right)$.

По деф. на $\Omega\left(n^2\right) = \left\{f \mid \exists\, c > 0\; \exists\, n_0 \in \mathbb{N}_0\; \forall\, n \geq n_0 \text{ е изпълнено } 0 \leq c.n^2 \leq f(n)\right\}$.

Тоест търсим константа $c > 0$ и търсим константа $n_0 \in \mathbb{N}_0$ такива, че ни е изпълнено $\forall\, n \geq n_0 : 0 \leq c.n^2 \leq T(n)$.

Нека $b = \min\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, ?\}$. [3]

Ще докажем, че за $c = b$ и за някое $n_0$ (ще установим по-надолу), че $\forall\, n \geq n_0 : T(n) \geq b.n^2$.

**База:**

Нека $k \in \{1, \ldots, n_{st} - 1\}$. Ще докажем, че $T(k) \geq b.k^2$.

От деф. на $b = \min\left\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, ?\right\} \overset{1 \leq k \leq n_{st}-1}{\Rightarrow} b \leq \frac{T(k)}{k^2} \Rightarrow T(k) \geq b.k^2$, което искахме да докажем.

**ИП:**

Нека допуснем, че е изпълнено $\forall\, m < n : T(m) \geq b.m^2$.

**ИС:**

Ще докажем, че е изпълнено за $n$, т.е $T(n) \geq b.n^2$.

$T(n) \overset{\text{def}}{=} T(n-1) + n \overset{\text{ИП}}{\geq} b(n-1)^2 + n = bn^2 - 2\,bn + b + n \overset{?}{\geq} bn^2$

$-2\,bn + b + n \overset{?}{\geq} 0$

$n(1 - 2\,b) + b \overset{?}{\geq} 0$

Може да забележим, че при $1 - 2\,b \geq 0$, горното неравенство ще е изпълнено за всяко $n \geq 0$. Оттук си избираме $\underbrace{n_0 \overset{(*)}{=} 1}_{\text{избираме 1, а не 0 защото нямаме } \frac{T(0)}{0^2} \text{ в } b}$, което не бяхме сигурни по-горе.

Остана да видим $1 - 2\,b \geq 0 \leftrightarrow 2\,b \leq 1 \leftrightarrow b \leq 1/2$. Това не сме си го подсигурили. Затова горе, където дефинирахме $b = \min\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, ?\}$, този въпросителен знак се въщаме и го заместваме с $1/2$.

Тоест в крайна сметка имаме $b = \min\left\{\frac{T(1)}{1^2}, \frac{T(2)}{2^2}, \frac{T(3)}{3^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, 1/2\right\}$.

И съответно показахме, че за $c = b$ & $n_0 = 1$, то $\forall\, n \geq \underbrace{n_0}_{1} : T(n) \geq c.n^2$. Оттук доказахме, че $T(n) = \Omega(n^2)$.

$T(n) = O(n^2)$ & $T(n) = \Omega(n^2) \Rightarrow T(n) = \theta(n^2)$, което искахме да докажем.

**[2]** Искаме $n_{st}$ да е достатъчно голямо число за да си гарантираме, че $n_{st} - 1 \geq \overbrace{n_0}^{\text{което избираме на местата означени със } (*)}$. Ако $n_{st} - 1 < n_0$, то тогава множеството $\{n_0, n_0 + 1, \ldots, n_{st} - 1\}$ би било празно .. т.е де факто ще сме без база.

**[3]** Винаги пишем $T(1)$ делено на "доказваната сложност". В случая искаме да докажем $n^2$ и затова $1^2$. Ако искахме $\log(n)$, то щяхме да пишем $T(2)$ върху $\log(2)$ и т.н. (не започваме от $T(1)$ върху $\log(1) = 0$ за да не разделим на нула). Съответно правим същото за $T(3), T(4), \ldots, T(n_{st} - 1)$. В края си пишем едно неизвестно '?' тъй като по-надолу в доказателството може да установим нагласена променлива, която ще ни трябва за да мине индукцията. В общия случай за $O$ пишем max, а за $\Omega$ пишем min.

**Заб** При $n_0 > 1$ може да видим, че всъщност в базата доказваме повече неща от необходимото.. всъщност в базата трябва да доказваме за $b = \max/\min\left\{\frac{T(n_0)}{n_0^2}, \ldots, \frac{T(n_{st}-1)}{(n_{st}-1)^2}, ?\right\}$. Но може да се съгласите, че в базата доказваме по силно твърдение. Тоест това не представлява проблем.

**Зад. 4** //Използвайте: развиване

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

Тук директно се вижда, че $T(n) = \frac{1}{n} + \frac{1}{n-1} + \ldots + \frac{1}{1} + \underbrace{T(0)}_{\text{const}} \overset{\text{хармоничен}}{\underset{\text{ред}}{=}} \theta(ln(n))$. Разбира се отново трябва формално док. с индукция.

**Зад. 5** //Използвайте: развиване

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

$$T(n) = 2\,T(n-1) + \frac{1}{n} = 2\,(\underbrace{2\,T(n-2) + \tfrac{1}{n-1}}_{T(n-1)}) + \frac{1}{n} = 4\,T(n-2) + \frac{2}{n-1} + \frac{1}{n} =$$

$$4\,(\underbrace{2\,T(n-3) + \tfrac{1}{n-2}}_{T(n-2)}) + \frac{2}{n-1} + \frac{1}{n} = 8\,T(n-3) + \frac{4}{n-2} + \frac{2}{n-1} + \frac{1}{n} = \ldots = 2^n\,\underbrace{T(0)}_{\text{const}} + \frac{2^{n-1}}{1} + \frac{2^{n-2}}{2} + \ldots + \frac{1}{n} = 2^n\,\underbrace{T(0)}_{\text{const}} + \underbrace{2^n \sum_{i=1}^{n} \frac{1}{i2^i}}_{O(2^n)} = \theta(2^n).$$

Разбира се отново трябва формално док. с индукция.

**Зад. 6** //Използвайте: развиване

$T(n) = \frac{n}{n+1}\,T(n-1) + 1$

$$T(n) = \frac{n}{n+1}\,(\underbrace{\tfrac{n-1}{n}\,T(n-2) + 1}_{T(n-1)}) + 1 =$$

$$\frac{n-1}{n+1}\,T(n-2) + \frac{n}{n+1} + \frac{n+1}{n+1} = \ldots = \frac{1}{n+1}\,\underbrace{T(0)}_{\text{const}} + \frac{2}{n+1} + \frac{3}{n+1} + \ldots + \frac{n+1}{n+1} = \theta(n^{-1}) + \frac{1}{n+1}\,(\underbrace{2 + 3 + \ldots + (n+1)}_{\theta(n^2)}) = \theta(n)$$

.Разбира се отново трябва формално док. с индукция.

**Зад. 7** //Използвайте: характеристично уравнение

$T(n) = T(n-2) + T(n-4) + \ldots + \underbrace{T(n\,\%2)}_{T(0) \vee T(1)}$

$T(n+2) = T(n) + \underbrace{T\,(n-2) + T(n-4) + \ldots + T(n\,\%2)}_{T(n)} = 2\,T(n)$

$x^2 = 2 \mapsto \left\{-\sqrt{2}\,,\ \sqrt{2}\right\}_M$

Оттук $T(n) = c_1\left(-\sqrt{2}\right)^n + c_2\left(\sqrt{2}\right)^n \overset{[4]}{=} \theta\left(\sqrt{2}\right)$.

**[4]** Разбираме се, че в рамките на текущия курс, няма как асимптотично да имаме отрицателно, тъй като семантиката на това рек. ур. е сложността на конкретен алгоритъм / програма.

**Зад. 8**

*a*) Намерета сложността на следната функция:

```
A(n) : // n ∈ ℕ⁺
    s ← 0
    for i ← 1 to n − 1
        s ← s + 2 A(i) + 1
    return s
```

Лесно се съобразява, че рек. уравнение, описващо сложността на този алгоритъм е следния: $T(n) = T(1) + T(2) + \ldots + T(n-1)$. Намираме асимптотиката както в задача 7.

$T(n+1) = T(n) + \underbrace{T\,(n-1) + \ldots + T(1)}_{T(n)} = 2\,T(n)$

$x = 2 \mapsto \{2\}_M$

Оттук $T(n) = c_1\,2^n = \theta(2^n)$.

*b*) Намерета сложността на следната функция:

```
B(n) : // n ∈ ℕ⁺
    s ← 0
    for i ← 1 to n − 1
        s ← s + B(i) + B(i) + 1
    return s
```

Лесно се съобразява, че рек. уравнение, описващо сложността на този алгоритъм е следния: $T(n) = 2\,T(1) + 2\,T(2) + \ldots + 2\,T(n-1)$. Намираме асимптотиката както в задача 7.

$T(n+1) = 2\,T(n) + \underbrace{2\,T(n-1) + \ldots + 2\,T(1)}_{T(n)} = 3\,T(n)$

$x = 3 \mapsto \{3\}_M$

Оттук $T(n) = c_1\,3^n = \theta(3^n)$.

**Зад. 9** //Използвайте: полагане

$T(n) = 2\,T\left(\sqrt{n}\right) + 1$

Полагаме: $n = 2^{2^m}$, $m = \log(\log(n))$.

Нека $S(m) = T\left(2^{2^m}\right) = T(n)$.

Тогава имаме: $S(m) = T\left(2^{2^m}\right) = 2\,T\left(2^{2^m/2}\right) + 1 = 2\,T\left(2^{2^{m-1}}\right) + 1 = 2\,S(m-1) + 1$

Пресятаме по вече познат ни метод $S(m) = \theta(2^m) = \theta\left(2^{\log(\log(n))}\right) = \theta(\log(n)) = T(n)$.

**Зад. 10** //Използвайте: полагане

$T(n) = T\left(\sqrt{n}\right) + 1$

Полагаме: $n = 2^{2^m}$, $m = \log(\log(n))$.

Нека $S(m) = T\left(2^{2^m}\right) = T(n)$.

Тогава имаме: $S(m) = T\left(2^{2^m}\right) = T\left(2^{2^{m-1}}\right) + 1 = S(m-1) + 1$

Пресмятаме по вече познат ни метод $S(m) = \theta(m) = \theta(\log(\log(n))) = T(n)$.

**Любопитно** Нека вземем едно произволно число. Да кажем 173. Нека го разгледаме в двоична бройна система $173_{(10)} = 10101100_{(2)}$. Нека вземем $\left\lfloor \sqrt{173} \right\rfloor = 13$. Нека него го вземем в двоична бройна система. $13_{(10)} = 1101_{(2)}$. Нека сега вземем $\left\lfloor \sqrt{13} \right\rfloor = 3 = 11_{(2)}$. И накрая да вземем $\left\lfloor \sqrt{3} \right\rfloor = 1 = 1_{(2)}$. Може да забележим, че когато коренуваме едно число, то броя на цифрите му в двоична бройна система намалява двойно. Тоест ако коренуваме едно число, докато не стигнем до $1_{(10)} = 1_{(2)}$, то ще трябва да го коренуваме $\log_2$(*дължина на числото в двоична бройна система*) на брой пъти. А колко е дължина на числото в двоична бройна система? Еми точно $\log_2$(*числото*). Тоест излезе, че ще направим общо $\log_2(\log_2(n))$ на брой стъпки, докато не стигнем $1_{(10)} = 1_{(2)}$. Това е хубава интерпретация на операцията коренуване. Ако някъде видите рекурентно уравнение, което отдясно има $T\left(\sqrt{n}\right)$, то най-вероятно ще има намесено $\log(\log(n))$.