

1. ZÁKLADNÍ METODY ŘEŠENÍ REKURENCÍ

1.1. PŘÍKLADY

Všude řešeno, rekurence je způsob, jaké má navození, kde
připadá, kde jsou již dříve řešeno to samé odstupem.

PŘÍKLAD 1.1.1 (PROBLÉM HANOJSKÝCH VĚŽÍ)

Je dáno tři tyče A, B, C, a n kroužků, kde je polezení
v pořadí, kroužků přeměnu na tyč A, zatímco ostatní dva jsou
je přeměnu (na začátku $n=4$).



Ukolem je přemístit všechny n A na B, zatímco přemístit C,
je přemístit, a v pořadí, kde je přemístit je přemístit
a nikdy není kroužků přemístit na kroužků menších přemístit.

ALGORITHMUS 1.1.2. (PROBLÉM HANOJSKÝCH VĚŽÍ - REKURZIVNÍ ALGORITHMUS)

1. Přemístit $n-1$ kroužků z A na C.
2. Přemístit největší kroužek z A na B.
3. Přemístit $n-1$ kroužků z C na B.

Klobouk algoritmus je rekurzivní. Před přemístit jednotlivých kroužků
ovšem $T(n)$. To $T(n)$ přemístit

$$T(n) = \begin{cases} 1 & \text{val } n=1 \\ 2T(n-1)+1 & \text{val } n>1 \end{cases}$$

Izrēķini pie skaitli 15 bāzi $T(n)$.

Kā jau redzējām, risinājumu problēmas Hanojā ir vērt. Cēl ir šķīst, mums ir dotas pārsūt burti n no A uz B un B uz A .

Vēlams pārsūt burti šādi pārsūtīt rekurīvi algoritmu, kurā pārsūt pārsūtīt burti n no A uz B un B uz A .
Izrēķini skaitli

$$T'(n) = \begin{cases} 2 & \text{val } n=1 \\ 3T'(n-1)+2 & \text{val } n>1 \end{cases}$$

- (I) kad $n-1$ burti pārsūt n no A uz B , to ir $T'(n-1)$ pārsūt pārsūtīt burti
- (II) pārsūt nākamā burti n no A uz C
- (III) pārsūtīt burti $n-1$ burti n no B uz A , to ir $T'(n-1)$ pārsūt pārsūtīt burti
- (IV) pārsūtīt nākamā burti n no C uz B
- (V) pārsūtīt burti $n-1$ burti n no A uz B , to ir $T'(n-1)$ pārsūt pārsūtīt burti

Algoritms skaidri rāda, ka rekurīvi risinājums, kas ir algoritms, mums ir šāds risinājums rodāt un pārsūt.

PIEKĻAID 1.1.3.

Kad mēs mēsme algoritmu risināt problēmu P mēsme $n = c^i$ (c, i ir daļskaitļi, $c \geq 2, i \geq 0$) tādā veidā, ka rekurīvi risinājums mums ir, kas $a, b_1, b_2, \dots, d$ ir burti burti (a ir mēsme cēl) =

1. Keď problém P , s časťmi b_1, m, m_1 a podproblémom
keď typ a riešenie $\frac{m}{c}$
2. Vyšetrujeme riešenie, s určitou časťou riešenia.
3. Keď, s časťmi b_2, m , riešenie problému P a riešenie riešenia je
podproblém.

Keď časť riešenia $T(n)$ vyjde ako algebraický výraz

$$T(n) = \begin{cases} d & \text{keď } n=1 \\ aT\left(\frac{n}{c}\right) + b_1n + b_2n & \text{keď } n>1 \end{cases}$$

1.2. SUBSTITUČNÁ METODA

Vždy je vhodné mať určitú metódu pre riešenie problému.
Keď časť riešenia riešenia.

PRÍKLAD 1.2.1.

Keď máme funkciu $T(n)$ (nie problém) a $T'(n)$ (nie
riešenie) riešenie

n	1	2	3	4	5	6	7	8
$T(n)$	1	3	7	15	31	63	127	255
$T'(n)$	2	8	26	80	242	728	2186	6560

Keď máme funkciu $T(n) = 2^n - 1$, $T'(n) = 3^n - 1$.
Keď máme funkciu $T(n) = 2^n - 1$ a $T'(n) = 3^n - 1$.
Keď máme funkciu $T(n) = 2^n - 1$ a $T'(n) = 3^n - 1$.
Keď máme funkciu $T(n) = 2^n - 1$ a $T'(n) = 3^n - 1$.

PRÍKLAD 1.2.2. (PROBLÉM HANOJSKÝCH VEŽÍ)

Keď máme funkciu $T(n) = 2^n - 1$ a $T'(n) = 3^n - 1$.

$$n=1, T(1) = 2^1 - 1 = 2 - 1 = 1, \text{ o.k.}$$

$$\begin{aligned} n > 1: 2T(n-1) + 1 &= 2 \cdot (2^{n-1} - 1) + 1 \\ &= 2 \cdot 2^{n-1} - 2 + 1 \\ &= 2^n - 1 \\ &= T(n) \quad \text{o.k.} \end{aligned}$$

1.3 ITERAČNÍ METODA

Každý rekursivní problém můžeme převést na součet, který můžeme řešit opakovaně či odhadnout.

PŘÍKLAD 1.3.1.

Každý rekursivní problém můžeme převést na součet, který můžeme řešit opakovaně či odhadnout.

$$\begin{aligned} T'(n) &= 3 \cdot T'(n-1) + 2 \\ &= 3 \cdot (3 \cdot T'(n-2) + 2) + 2 \\ &= 3^2 \cdot T'(n-2) + 3 \cdot 2 + 2 \\ &= 3^2 \cdot (3 \cdot T'(n-3) + 2) + 3 \cdot 2 + 2 \\ &= 3^3 \cdot T'(n-3) + 3^2 \cdot 2 + 3 \cdot 2 + 2 \\ &\vdots \\ &= 3^{n-1} \cdot T'(n - (n-1)) + 3^{n-2} \cdot 2 + \dots + 3 \cdot 2 + 2 \\ &= 3^{n-1} \cdot T'(1) + 3^{n-2} \cdot 2 + \dots + 3 \cdot 2 + 2 \\ &= 3^{n-1} \cdot 2 + 3^{n-2} \cdot 2 + \dots + 3 \cdot 2 + 2 \\ &= (3^{n-1} + 3^{n-2} + \dots + 3 + 1) \cdot 2 \end{aligned}$$

$$\begin{aligned} \cancel{3^n} + \cancel{3^{n-1}} + \dots + \cancel{3} + 1 &= x \quad | \cdot 3 \\ \cancel{3^n} + \cancel{3^{n-1}} + \dots + \cancel{3} + 3 &= 3x \\ \hline 3^n - 1 &= 2x \\ \frac{3^n - 1}{2} &= x \end{aligned}$$

$$T'(n) = \frac{3^n - 1}{2} \cdot 2 = 3^n - 1, \quad \underline{\underline{T'(n) = 3^n - 1}}$$

PIKILAD 1.2.2.

kur rekurs

$$T(n) = \begin{cases} b & \text{for } n = 1 \\ a \cdot T\left(\frac{n}{c}\right) + d_n & \text{for } n = c^i > 1 \end{cases}$$

(kur $b = d_1 + d_2$ e pēdējais, $n = c$) ~~rekurs~~ ^{replac} ~~rekurs~~
analīzē algoritmu laiku a priekš dāvanu

$$\begin{aligned} T(n) &= a T\left(\frac{n}{c}\right) + d_n \\ &= a \left(a T\left(\frac{n}{c^2}\right) + b \frac{n}{c} \right) + d_n \\ &= a^2 T\left(\frac{n}{c^2}\right) + a b \frac{n}{c} + d_n \\ &= a^2 \left(a T\left(\frac{n}{c^3}\right) + b \frac{n}{c^2} \right) + a b \frac{n}{c} + d_n \\ &= a^3 T\left(\frac{n}{c^3}\right) + a^2 b \frac{n}{c^2} + a b \frac{n}{c} + d_n \\ &\vdots \\ &= a^i T\left(\frac{n}{c^i}\right) + a^{i-1} b \frac{n}{c^{i-1}} + \dots + a^2 b \frac{n}{c^2} + a b \frac{n}{c} + d_n \\ &= a^i T(1) + a^{i-1} b \frac{n}{c^{i-1}} + \dots + a^2 b \frac{n}{c^2} + a b \frac{n}{c} + d_n \\ &= a^i \cdot b + a^{i-1} \cdot b \cdot \frac{n}{c^{i-1}} + \dots + a^2 b \frac{n}{c^2} + a b \frac{n}{c} + d_n \\ &= a^i \cdot b \cdot \frac{n}{c^i} + a^{i-1} \cdot b \cdot \frac{n}{c^{i-1}} + \dots + a^2 b \frac{n}{c^2} + a b \frac{n}{c} + d_n \\ &= b_n \left(\left(\frac{a}{c}\right)^i + \left(\frac{a}{c}\right)^{i-1} + \dots + \left(\frac{a}{c}\right)^2 + \frac{a}{c} + 1 \right) \\ &= b_n \cdot \sum_{0 \leq j \leq i} \left(\frac{a}{c}\right)^j \\ &= b_n \cdot \sum_{0 \leq j \leq \log_{\frac{a}{c}} n} \left(\frac{a}{c}\right)^j \end{aligned}$$

(6)

$a, c \in \mathbb{Z}, a \geq 1, c \geq 2$
 lecture 3 reply:

(I) $a < c$:

$$T(n) \leq cn \cdot \sum_{0 \leq j} \left(\frac{a}{c}\right)^j$$

As $0 < \frac{a}{c} < 1$, then $\sum_{0 \leq j} \left(\frac{a}{c}\right)^j$ converges, $\sum_{0 \leq j} \left(\frac{a}{c}\right)^j = \Delta, \Delta > 0$.

$$T(n) \leq cn \cdot \Delta = O(\Delta \cdot n)$$

$$T(n) \geq cn \cdot \left(\frac{a}{c}\right)^0 = cn$$

$$cn \leq T(n) \leq (c\Delta)n, \quad \boxed{T(n) = \Theta(n)}$$

(II) $a = c$:

$$\sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j = \sum_{0 \leq j \leq \log_c n} 1 = 1 + \log_c n$$

$$\log_c n \leq 1 + \log_c n \leq 2 \log_c n$$

As $n \geq c$: $\log_c n \leq \sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j \leq 2 \log_c n$

$$cn \log_c n \leq cn \sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j \leq 2cn \log_c n$$

$$cn \log_c n \leq T(n) \leq 2cn \log_c n$$

$$\log_c n = \frac{\log n}{\log c}, \quad \log_c n = \log$$

$$\log_{10} m = \frac{\log_c m}{\log_c 10}, \quad \log_c m = \log_c 10 \cdot \log_{10} m$$

$$\log_c m = \log_c 10 \cdot \log m$$

$$1.5 m \geq c: \text{ then } \log_c 10 \log m \leq T(m) \leq 2.0 m \log_c 10 \cdot \log m$$

$$(1.5 \log_c 10) \cdot m \cdot \log m \leq T(m) \leq (2.0 \log_c 10) m \cdot \log m$$

$$\cancel{T(m) \leq m \cdot \log}$$

$$\boxed{T(m) = \Theta(m \cdot \log m)}$$

(II) $a > c$:

$$\sum_{0 \leq j \leq \log_c m} \left(\frac{a}{c}\right)^j \geq \left(\frac{a}{c}\right)^{\log_c m} = \frac{a^{\log_c m}}{c^{\log_c m}} = \frac{a^{\log_c m}}{m} = \frac{m^{\log_c a}}{m}$$

$$\log_a m = \frac{\log_c m}{\log_c a}, \quad \log_c a \cdot \log_a m = \log_c m$$

$$a^{\log_c m} = a^{\log_c a \cdot \log_a m} = (a^{\log_a m})^{\log_c a} = m^{\log_c a}$$

$$\sum_{0 \leq j \leq \log_c m} \left(\frac{a}{c}\right)^j \geq \frac{m^{\log_c a}}{m}$$

$$1 + q + q^2 + \dots + q^k = X$$

$$\underline{q + q^2 + q^3 + \dots + q^{k+1} = qX}$$

$$q^{k+1} - 1 = qX - X$$

$$q^{k+1} - 1 = X(q-1)$$

$$X = \frac{q^{k+1} - 1}{q - 1} \quad (15 \quad q \neq 1)$$

by $q = \frac{a}{c}$, ~~where~~ $k = \log_c n$ where

$$\sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j = \frac{\left(\frac{a}{c}\right)^{\log_c n + 1} - 1}{\frac{a}{c} - 1} \leq \frac{\left(\frac{a}{c}\right)^{\log_c n + 1}}{\frac{a}{c} - 1}$$
$$= \frac{\frac{a}{c} \cdot \left(\frac{a}{c}\right)^{\log_c n}}{\frac{a}{c} - 1} = \frac{\frac{a}{c} \cdot \frac{n^{\log_c a}}{n}}{\frac{a}{c} - 1} = \frac{\frac{a}{c}}{\frac{a}{c} - 1} \cdot \frac{n^{\log_c a}}{n}$$

$$\sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j \leq \frac{\frac{a}{c}}{\frac{a}{c} - 1} \cdot \frac{n^{\log_c a}}{n}$$

$$\frac{n^{\log_c a}}{n} \leq \sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j \leq \frac{\frac{a}{c}}{\frac{a}{c} - 1} \cdot \frac{n^{\log_c a}}{n}$$

$$l \cdot n^{\log_c a} \leq l \cdot n \sum_{0 \leq j \leq \log_c n} \left(\frac{a}{c}\right)^j \leq l \cdot \frac{\frac{a}{c}}{\frac{a}{c} - 1} \cdot n^{\log_c a}$$

$$l \cdot n^{\log_c a} \leq T(n) \leq \left(l \cdot \frac{\frac{a}{c}}{\frac{a}{c} - 1} \right) \cdot n^{\log_c a}$$

$$\boxed{T(n) = \Theta(n^{\log_c a})}$$

Vă reaminteți că, în cazul algoritmului recursiv a pașilor recursivi
pe care îl prezintă $\frac{a}{c}$, a nu se poate aplica, deși reținem, că
nu reprezintă numărul, deși puterile, deși nu putem aplica, se
deosebesc a luatei numărului pe care îl analizăm (și trebuie
cunoscute puterile lui c).
 $\left[\log_c \frac{a}{c} = \log_c a - \log_c c = \log_c a - 1, \log_c a = 1 + \log_c \frac{a}{c} \right]$

1.4 REDUKCE NA ALGEBRAICKÉ ROVNICE

Veliká třída rovnic, viz. homogenní lineární rovnice, může být vyřešena redukcí na algebraické rovnice. Vhodné pro to obecnou metodu, kterou následně vyřídíme v příkladech.

PŘÍKLAD 1.4.1. (FIBONACCIHO ČÍSLA)

Leonardo Fibonacci v roce 1202 popsal poprvé číselnou posloupnost

$$F_0 = 0, F_1 = 1 \quad (\text{počáteční podmínky})$$

$$F_n = F_{n-1} + F_{n-2} \quad \text{pro } n \geq 2 \quad (\text{rekurentní rovnice})$$

Fibonacciho čísla jsou přesně a vyjádřené jejich poprvé principem čísel:

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, \dots$$

Je možné se ptát, zda je možné něco říci o minimálním F_n je větší než F_i pro nějaké i , $0 \leq i < n$.
Tedy: můžeme najít explicitní vzorec pro F_n ?

Pokusme se najít nějaké řešení, které bude mít tvar $F_n = r^n$, kde r je nějaká reálná hodnota.
Následně je r^n rovnice rekurentní rovnice. Pak

$$r^n = r^{n-1} + r^{n-2}$$

Pro nějaké r a $n \geq 2$. Jde-li o $r \neq 0$ (pro $r=0$ dostaneme pouze nulovou posloupnost):

$$r^n = r^{n-1} + r^{n-2} \quad | : r^{n-2}$$

$$r^2 = r + 1$$

Thì việc nó đã xong!

$$r^2 - r - 1 = 0, \quad D = (-1)^2 - 4 \cdot 1 \cdot (-1) = 1 + 4 = 5,$$

$$r_1 = \frac{1 + \sqrt{5}}{2}, \quad r_2 = \frac{1 - \sqrt{5}}{2}$$

Thì ta có hai r_1^n, r_2^n nghiệm riêng của phương trình.

Để tìm dãy số ta tìm dãy số $\lambda r_1^n + \mu r_2^n$ thỏa mãn điều kiện ban đầu:

$$\begin{aligned} & (\lambda r_1^{n-1} + \mu r_2^{n-1}) + (\lambda r_1^{n-2} + \mu r_2^{n-2}) \\ &= \lambda (r_1^{n-1} + r_1^{n-2}) + \mu (r_2^{n-1} + r_2^{n-2}) \\ &= \lambda r_1^{n-2} (r_1 + 1) + \mu r_2^{n-2} (r_2 + 1) \\ &= \lambda r_1^{n-2} \cdot r_1^2 + \mu r_2^{n-2} \cdot r_2^2 \\ &= \lambda r_1^n + \mu r_2^n \end{aligned}$$

Thì ta sẽ chọn λ và μ sao cho, dãy số thỏa mãn điều kiện ban đầu, hay:

$$\lambda r_1^0 + \mu r_2^0 = 0, \quad \lambda r_1^1 + \mu r_2^1 = 1$$

$$\text{thì ta có } F_n = \lambda r_1^n + \mu r_2^n.$$