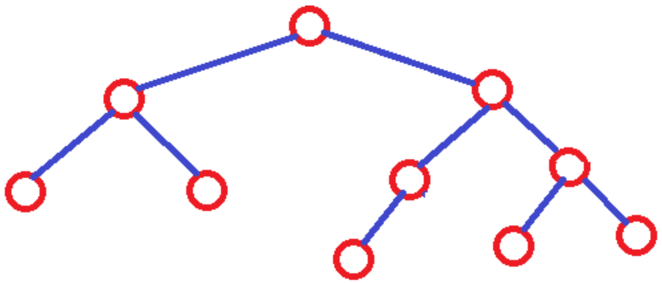
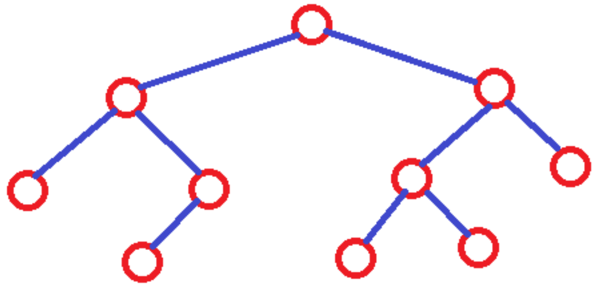
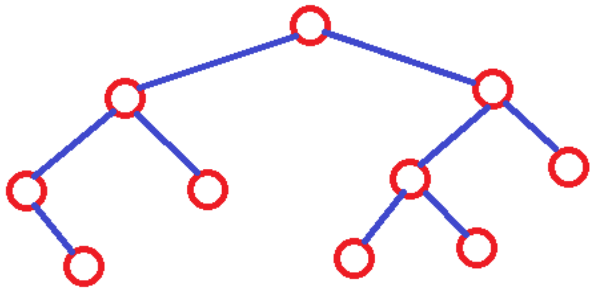
MÁME PROBLÉM

Kolik existuje BVS o 10 vrcholech?







Vrcholy (uzly) mají hodnoty
1, 2, 3, 4, 5, 6, 7, 8, 9 a 10
a jsou uspořádané (rostou)
odleva doprava.

Rozdělení dětí do skupin

Na hřišti je 6 dětí,

kolika způsoby se mohou rozdělit do 3 skupin?

Skupiny nemají žádné jméno,

jde jen o to, kdo si s kým bude hrát.

Mrkev a petržel

- Máme 10 záhonků
- na každý přijde buďto mrkev nebo petržel
- nikdy nesmí být dva záhonky s mrkví vedle sebe

Kolika způsoby můžeme záhonky osázet/osít?

Mrkev a petržel

Krok 1: Úlohu zobecníme

Z_k // počet způsobů celkem

$=$ M_k // počet způsobů když poslední je mrkev

$+$ P_k // počet způsobů když poslední je petržel

Mrkev a petržel

Krok 1: Úlohu zobecníme

Z_k // počet způsobů celkem

$= M_k$ // počet způsobů když poslední je mrkev

$+ P_k$ // počet způsobů když poslední je petržel

Krok 2: Určíme triviální případy

$M_1 = P_1 = 1$

Mrkev a petržel

Krok 1: Úlohu zobecníme

Z_k // počet způsobů celkem

= M_k // počet způsobů když poslední je mrkev

\+ P_k // počet způsobů když poslední je petržel

Krok 2: Určíme triviální případy

$M_1 = P_1 = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

$M_k = \ldots \qquad P_k = \ldots$

Mrkev a petržel

Krok 1: Úlohu zobecníme

Z_k // počet způsobů celkem

$=$ M_k // počet způsobů když poslední je mrkev

$+$ P_k // počet způsobů když poslední je petržel

Krok 2: Určíme triviální případy

$M_1 = P_1 = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

$M_k = P_{k-1}$ $\qquad$ $P_k = M_{k-1} + P_{k-1}$

Mrkev a petržel

Krok 1: Úlohu zobecníme

Z_k // počet způsobů celkem

= M_k // počet způsobů když poslední je mrkev

\+ P_k // počet způsobů když poslední je petržel

Krok 2: Určíme triviální případy

$M_1 = P_1 = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

$M_k = P_{k-1}$ $\quad$ $P_k = M_{k-1} + P_{k-1}$

Krok 4: Spočteme, co potřebujeme

// ukládáme si (mezi)výsledky, hodí se na to tabulkový procesor

Rozdělení dětí do skupin

Krok 1: Úlohu zobecníme

$P_{D,\ S}$ // počet způsobů rozdělení D dětí do S skupin

Rozdělení dětí do skupin

Krok 1: Úlohu zobecníme

$P_{D,\,S}$ // počet způsobů rozdělení D dětí do S skupin

Krok 2: Určíme triviální případy

$P_{D,\,1} = \ldots \qquad P_{D,\,D} = \ldots$

Rozdělení dětí do skupin

Krok 1: Úlohu zobecníme

$P_{D,\,S}$ // počet způsobů rozdělení D dětí do S skupin

Krok 2: Určíme triviální případy

$P_{D,\,1} = 1$ $P_{D,\,D} = 1$

Rozdělení dětí do skupin

Krok 1: Úlohu zobecníme

$P_{D,\,S}$ // počet způsobů rozdělení D dětí do S skupin

Krok 2: Určíme triviální případy

$P_{D,\,1} = 1$ $\quad$ $P_{D,\,D} = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

Rozdělení dětí do skupin

Krok 1: Úlohu zobecníme

$P_{D,\,S}$ // počet způsobů rozdělení D dětí do S skupin

Krok 2: Určíme triviální případy

$P_{D,\,1} = 1 \qquad P_{D,\,D} = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

Tip: Položit si nějakou šikovnou otázku:

Co se stane, když přijde poslední dítě?

Rozdělení dětí do skupin

Krok 1: Úlohu zobecníme

$P_{D,\ S}$ // počet způsobů rozdělení D dětí do S skupin

Krok 2: Určíme triviální případy

$P_{D,\ 1} = 1 \qquad P_{D,\ D} = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

Tip: Položit si nějakou šikovnou otázku:

Co se stane, když přijde poslední dítě?

$P_{D,\ S} = P_{D-1,\ S} * S$ // přidá se do některé skupiny

$+ P_{D-1,\ S-1}$ // založí si vlastní skupinu

Krok 4: Spočteme, co potřebujeme

Kolik existuje BVS o 10 vrcholech?

Krok 1: Úlohu zobecníme

P_K // počet BVS o K vrcholech

Krok 1: Úlohu zobecníme

P_K // počet BVS o K vrcholech

Krok 2: Určíme triviální případy

$P_0 = 1$ $\qquad$ $P_1 = 1$

Kolik existuje BVS o 10 vrcholech?

Krok 1: Úlohu zobecníme

P_K // počet BVS o K vrcholech

Krok 2: Určíme triviální případy

$P_0 = 1$ $P_1 = 1$

Krok 3: Určíme rekurentní vzorec/vzorce

Tip: Položit si nějakou šikovnou otázku:

Co když je kořenem i-tý prvek: => $P_{(i-1)} * P_{(K-i)}$

Kolik existuje BVS o 10 vrcholech?

$$P(K)=\sum_{i=1}^{K} P_{i-1} \times P_{K-i}$$

Jak přijdeme na ty kouzelné vzorečky?

Napíšeme si rekurzivní funkci…

...a přidáme jí paměť (cache),

aby neopakovala stejné výpočty.

(„Memoizace“)

(Kouzelné vzorečky pak nepotřebujeme,

můžeme nechat tu funkci.)

Úlohy „Kolik celkem?“

versus

úlohy „Kolik nejlépe (a jak)?“

Nutná podmínka:

Optimální řešení obsahuje
jako své části
optimální řešení menších úloh.

Násobení matic

1) Na vynásobení dvou matic rozměrů

   a×b a b×c

   potřebujeme a×b×c operací násobení.
2) Násobení matic je asociativní.
3) Na různá pořadí potřebujeme různý počet násobení

(1×8) ((8×1)(1×5)) » 8×1×5 + 1×8×5 = 40+40 = 80

((1×8)(8×1)) (1×5) » 1×8×1 + 1×1×5 = 8+5 = 13

Kolik (nejméně) operací násobení potřebujeme k vynásobení řady matic známých rozměrů?

Násobení matic

Příklad:

4 matice, jejich rozměry jsou

(2×8) (8×4) (4×5) (5×2)

Vstup jsou rozměry matic:

[2,8,4,5,2], s indexy 0..4. = R_0, R_1, R_2, R_3, R_4

Krok 1: Úlohu zobecníme:

$P_{k,i}$ // počet pro součin k matic, počínaje i-tou

Krok 2: Určíme triviální případy

$P_{1,i} = 0$

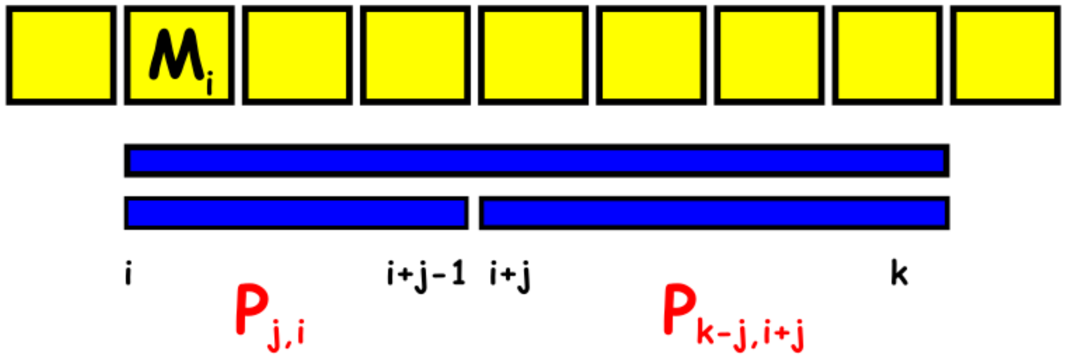
Násobení matic

Krok 3: Určíme rekurentní vzorec/vzorce

Jak najít nejlepší způsob vynásobení k matic?

Projedeme všechny způsoby,
jak součin k matic rozdělit na součin dvou matic
a vybereme z nich ten nejvýhodnější (minimální).

$P_{k,i} = \min(\quad P_{j,i} + P_{k-j,i+j} + R_i \times R_{i+j} \times R_{i+k} \quad)$, přes $j=1..k-1$



Dynamické programování

Řešení úlohy skládáme z řešení menších pod-úloh (která jsme si předtím vypočetli a zapamatovali).

Rozděl a panuj

Trochu to připomíná metodu Rozděl a panuj, ale liší se v tom, že u Rozděl a panuj víme, kde máme rozdělit, zatímco u dynamického programování to nevíme a tak zkoušíme všechny možnosti a z nich vybíráme nejlepší.

Memoizace

Rekurze s ukládáním výsledků,
když se vynechá ta rekurze, zbyde dynamické pgm.

Další úlohy

//Editační vzdálenost

//Nejdelší cesta v topologicky uspořádaném grafu

//Počet cest v topologicky uspořádaném grafu

Hry – počet a jak hrát…

//Optimální BVS…

Jak dát mat králem a věží ?