

Algoritmické strategie II

I.Kolingerová

Obsah:

4. Rozděl a panuj

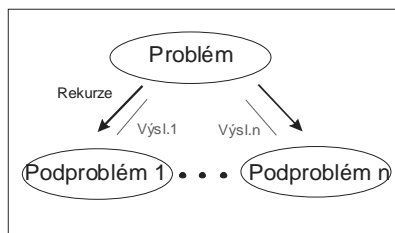
4. Rozděl a panuj



- Složitý problém rozdělit na jednodušší podproblémy
- Řešení podproblémů spojit na celkové řešení
- Pozor – řeší to původní problém?
- Pro dekomponovatelné problémy
- Dělení a spojování nesmí být dražší než původní problém
- Dekompozice a zejména spojování často obtížné

2

- Obvykle rekurze
- Původně vojenská strategie
- Příklad: mnoho významných, ale obvykle obtížných algoritmů, například mergesort, FFT, Strassenovo matic. násobení ...



3

Divide_and_conquer (P)

Input: Problém P velikosti n

Output: Řešení S problému P

- Step 1: **if** $n < R$ **then** Řeš P (\Rightarrow řešení S)
else begin
- Step 2: Rozděl P na podprob. P_1, \dots, P_k
- Step 3: **for** $i := 1$ **to** k **do**
 Divide_and_conquer (P_i);
- Step 4: Zkombinuj S_i do celk. řešení
end

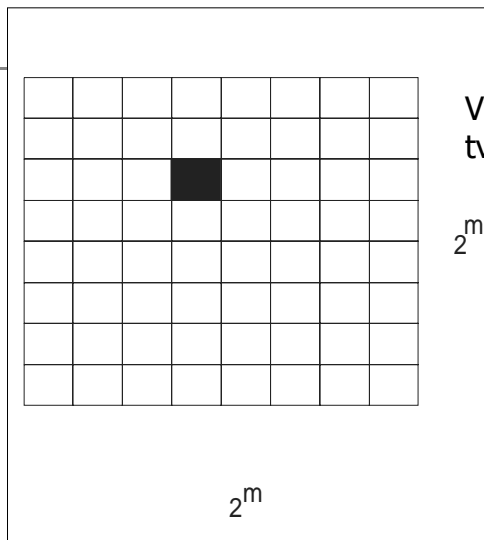
4

	<p style="text-align: right;">D&C strategie pro sort</p> <p>Př.1: Mergesort</p>
	<ul style="list-style-type: none"> ■ Rozdělujeme čísla do 2 sloupečků o polovičním počtu prvků, řadíme je, pak oba setříděné sloupečky zkombinujeme ■ Příklad: Při spojování $A=\{5,7,12,19\}$, $B=\{4,6,13,15\}$ musí min. být na začátku jednoho ze seznamů; opakujeme, spojujeme seřazené seznamy v $O(n)$ ■ MergeSort ($A[1,n]$) \sim <ul style="list-style-type: none"> Merge (MergeSort ($A[1, \lfloor n/2 \rfloor]$), (MergeSort ($A[\lfloor n/2 \rfloor + 1, n]$))) <p>Celkem $\log n$ úrovní, $O(n)$ na 1 úroveň $\Rightarrow O(n \log n)$</p> <p style="text-align: right;">5</p>

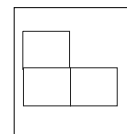
	<p style="text-align: right;">D&C strategie pro sort</p> <p>Př.2: Quicksort</p>
	<p>Úkol: najít a zopakovat</p> <p style="text-align: right;">6</p>

Př.3: Dlažďění šachovnice

D&C strategie
pro tiling



Vyplňte dlaždicemi
tvaru L



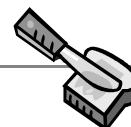
(beze zbytku,
lib. orientace)

7

Př.4: Binary search (půlení intervalu)

D&C strategie pro search

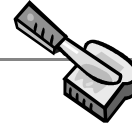
- Jak rychlé je půlení intervalu?
- Př.: dětská hra – 1 hráč vybere slovo, 2. opakovaně klade T/F otázky, snaží se slovo uhodnout. Pokud po 20 pokusech neuhodne, vyhrává 1.
- Vyhraje 1. vždycky/někdy/nikdy?



8

Př.4: Binary search, bisection (půlení intervalu) D&C strategie pro search

- Jak rychlé je půlení intervalu?
- Př.: dětská hra – 1 hráč vybere slovo, 2. opakovaně klade T/F otázky, snaží se slovo uhodnout. Pokud po 20 pokusech neuhodne, vyhrává 1.
- Vždy existuje vítězná strategie pro 2. hráče – otevře slovník v polovině, ptá se, zda hledané slovo abecedně před nebo za slovem ze slovníku ...
- Na 50 000 až 200 000 slov ze slovníku stačí 20 otázek !!



9

Př.5: V poli A je skupina 0, pak skupina 1, chceme vědět, kde je přechod D&C strategie pro search

0	0	0	0	0	1	1	1
---	---	---	---	---	---	---	---

- Pokud známe délku pole, půlíme interval
- Co když ne?
- Testujeme delší a delší intervaly, až narazíme na 1 => máme "okno", které půlíme – jednostranné binární vyhledávání
- Hlavně vhodné, když klíč někde blízko od dané pozice

10

Př.6: Rychlá mocnina

D&C strategie pro mocninu

- $a^n, n \gg$
- $n = (n \text{ div } 2) + (n+1) \text{ div } 2$
- pro n sudé: $a^n = (a^{n/2})^2$
- pro n liché: $a^n = a(a^{n \text{ div } 2})^2$

- $O(\log n)$ násobení
- \Rightarrow význam rovnoměrnosti dělení problému

- Úkol: napište jako funkci v prog.jazyce



11

Př.7: Odmocnina

D&C strategie pro odmocninu

- $\sqrt{n}, n \geq 1$ – výsledek nejméně 1 a nejvýše n
 $\Rightarrow l=1, r=n, m=(l+r) \text{ div } 2$
- srovnat m^2 s n :
 $n \geq m^2 \Rightarrow \sqrt{n} > m \Rightarrow l=m$
 $n < m^2 \Rightarrow \sqrt{n} < m \Rightarrow r=m$

- $O(\log n)$ kroků, pak jsme ve vzdál. ± 1 od \sqrt{n}

12