

# IAL projekt

## Varianta č.6 - Obarvení grafu

Adámek Josef  
Barnová Diana  
Vanický Jozef  
Weigel Filip

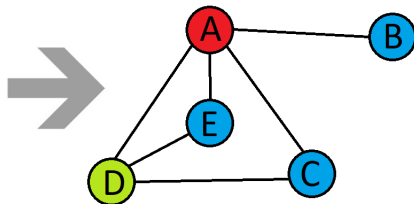
xadame42  
xbarno00  
xvanic09  
xweige01



12. prosince 2018

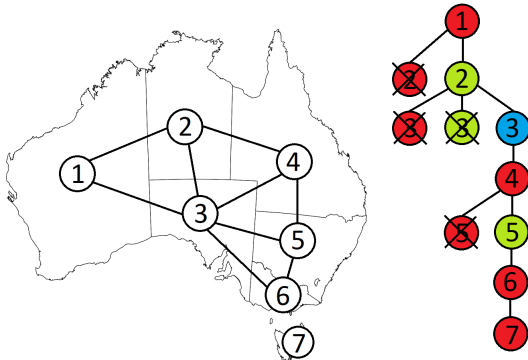
- problém: naleznout jedno z minimálních možných obarvení grafu
- použití neorientovaných grafů bez smyček a více hran mezi uzly

	A	B	C	D	E
A	0	1	1	1	1
B	1	0	0	0	0
C	1	0	0	1	0
D	1	0	1	0	1
E	1	0	0	1	0



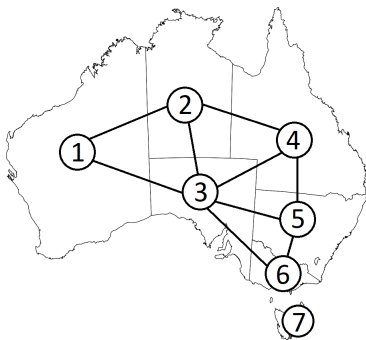
## Slepé prohledávání se zpětným navracením (backtracking):

- pro CSP (problémy s omezujícími podmínkami) je to metoda **úplná i optimální**
- navracení implementováno pomocí zásobníku
- bez vylepšení pomalá metoda pro optimální barvení grafů...



## Kontrola dopředu (forward checking):

- v podstatě rozšíření metody backtracking
- metoda je schopná dříve odhalit "větve" kombinací barev, které nevedou ke správnému řešení
- použití množin legálních barev uzlů
- implementována pomocí rekurze



3. kolo:

1: Č

2: Č Z M

3: Č Z M

4: Č Z M

5: Č Z M

6: Č Z M

7: Č Z M

- zvolili jsme matici sousednosti, protože naše úloha řeší i hustě spojené grafy
- množiny legálních barev implementovány pomocí polí typu bool

matice sousednosti

true	false	true	...	false
0	1	2		n*n

pole uzlů

			...	
0	1	2		n

Uzel

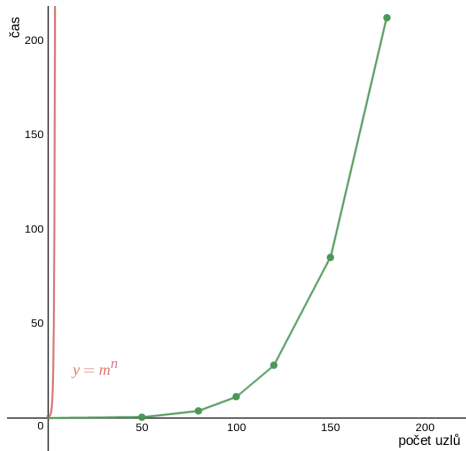
id
barva
množina barev

množina barev

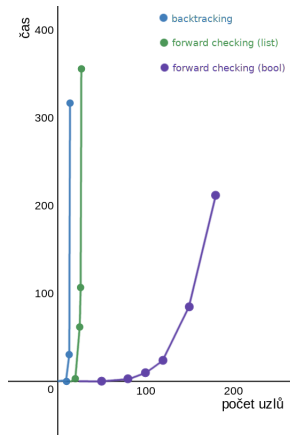
true	false	true	...	false
0	1	2		n

- hledání chromatického čísla grafu je NP-úplný problém
- časová složitost metody forward checking:  $O(m^n)$

počet uzlů	čas
50	0.3596 s
80	3.7076 s
100	11.2404 s
120	27.9151 s
150	85 s
180	212 s



- backtracking
- forward checking s množinami barev ve dvojsměrném seznamu
- forward checking s množinami barev v poli typu bool



- pravidelné schůzky, komunikace přes Facebook
- verzování programu pomocí nástrojů Git a GitHub
- testování v duchu metody Test-Driven Development



Děkujeme za pozornost