PAL: 6. cvičení

Tomáš Sieger

29. 10. 2020

Opakování z minula

Př. 5/12: (ne)izomorfní grafy

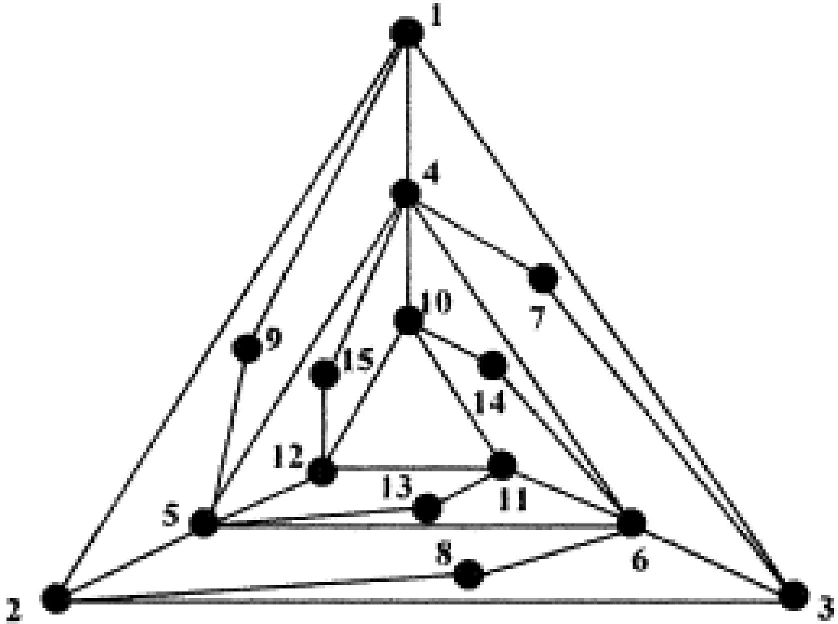
Platí tvrzení, že každé dva souvislé grafy se stejným počtem vrcholů, kde všechny vrcholy mají stejný stupeň, jsou izomorfní? (Tvrzení dokažte, nebo najděte protipříklad.)

Př. 5/4: počet bijekcí

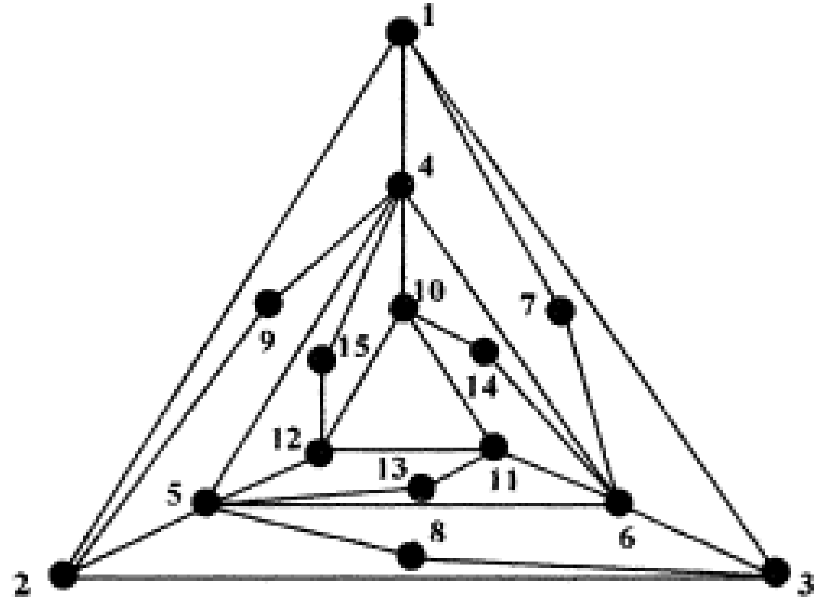
Máme dány dva neorientované grafy, každý obsahuje právě n uzlů a oba grafy mají skóre
$(n-1, n-2, n-3, n-4, ..., n/2+1, n/2, n/2, n/2-1, n/2-2, ..., 3, 2, 1)$,
to jest skoro všechny uzly grafu mají navzájem různý stupeň, s výjimkou dvou uzlů, které mají stejný stupeň $n/2$. Jaká bude asymptotická složitost ověření izomorfizmu těchto dvou grafů v závislosti na hodnotě n?

Př. 5/6: izomorfismus

Popište, jak budete co nejefektivněji rozhodovat, zda dva uvedené grafy jsou nebo nejsou izomorfní.



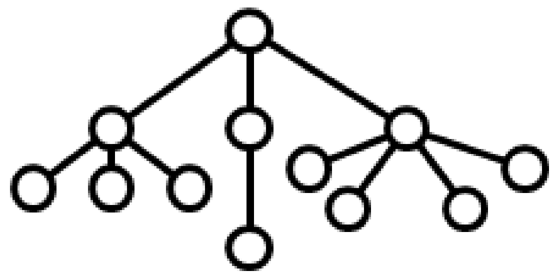
(a)



(b)

Př. 5/7d: tvorba certifikátu

Po sestavení certifikátu stromu odpovídá každému uzlu stromu určitý podřetězec konečného certifikátu. Sestavte certifikát daného stromu a určete, které jeho podřetězce odpovídají jednotlivým uzlům stromu.



Př. 5/8c: rekonstrukce stromu z certifikátu

Rekonstruujte strom z certifikátu:

c) 000101100101100010111

Př. 5/8d: rekonstrukce stromu z certifikátu

Rekonstruujte strom z certifikátu:

d) 000001011100111000010111

Kombinatorické algoritmy

Př. 6/1: podmnožiny písmen

Je dána množina malých písmen $P = \{a, b, c, \ldots, z\}$ bez diakritiky. Máme vygenerovat tabulku T s 8 sloupci, jejíž každý řádek bude obsahovat jednu 8-prvkovou podmnožinu množiny P tak, že každá buňka na řádku bude obsahovat právě jeden znak. Tabulka bude obsahovat všechny možné navzájem různé 8 prvkové podmnožiny P a žádná podmnožina se v T nebude opakovat. Určete, jak bude tabulka velká a zda ji váš počítač bude moci vyplnit během 1 sec.

Př. 6/2: všechny podmnožiny

Napište pseudokód funkce, která vypíše všechny neprázdné podmnožiny množiny $\{0, 1, 2, \ldots, n-1\}$.

Př. 6/3: přívětivé permutace

Mějme permutace množiny $M = \{1, 2, 3, \ldots, n\}, n > 4$. Permutaci p této množiny prohlásíme za přívětivou, pokud platí:

- ☐ $p(3) \in \{3, n\}$,
- ☐ $p(n) \in \{3, n\}$,
- ☐ $p(1) = 1$,
- ☐ $p(2) = 2$,
- ☐ $p(i) \in \{4, ..., n-1\}$ pro $i = 4, ..., n-1$.

Určete počet přívětivých permutací množiny M.

Př. 6/4: Grayův kód

Předpokládejme, že každý prvek Grayova code G_n, jímž je n-tice nul a jedniček, bude uložen v poli znaků o délce n. Napište pseudokód rekurzivní funkce, která pro dané n vygeneruje a vypíše celý Grayův code G_n.

Př. 6/5: číslování permutací

Všechny permutace množiny M s 98 prvky očíslujeme pořadovými čísly od 0 do $98! - 1$. V programu pak nepracujeme s permutacemi, ale jen s jejich pořadovými čísly. Víme, že budeme zkoumat pokaždé najednou 100 permutací, čili v paměti budeme muset mít uloženo právě 100 pořadových čísel různých permutací množiny M. Kolik minimálně bitů si musíme v paměti rezervovat, abychom si těchto 100 reprezentací mohli uložit?

Př. 6/6: předchozí podmnožina

Uvažujme všechny k-prvkové podmožiny množiny $M = \{1, 2, 3, ..., n\}, 1 \leq k \leq n$. Vyjděte z algoritmu transformujícího seznam prvků jedné podmnožiny na seznam prvků podmožiny bezprostředně následující v lexikografickém uspořádání těchto podmnožin. Navrhněte a popište algoritmus, který bude transformovat seznam prvků jedné podmnožiny na seznam prvků podmožiny bezprostředně předcházející v témže lexikografickém uspořádání. Bude mít stejnou asymptotickou složitost?

Př. 6/7: cykly permutací

Uvažujeme permutace množiny $M = \{1, 2, 3, \ldots, n\}$. Cyklus délky k v permutaci p definujeme jako množinu $A = \{a_1, a_2, ..., a_k\} \subset M$, pro kterou platí: $1 \leq a_1 < a_2 < \ldots < a_k \leq n$, $p(a_j) = a_j + 1$ pro $1 \leq j < k, p(a_k) = a_1$. Určete, kolik je takových permutací množiny M, které obsahují právě dva cykly, z nichž jeden má délku 4 a druhý délku $n - 4$.

Př. 6/8: pořadí permutace

Rank permutace π množiny $N = \{0, 1, 2, \ldots, n-1\}$ je pořadové číslo této permutace v seznamu všech permutací množiny N uspořádaném v rostoucím lexikografickém pořadí, přičemž prvky seznamu jsou číslovány od 0. Napište pseudokód funkce která v čase úměrném n vytiskne takovou permutaci π množiny N, jejíž rank je právě $n!/2$. Předpokládáme $n \geq 2$.