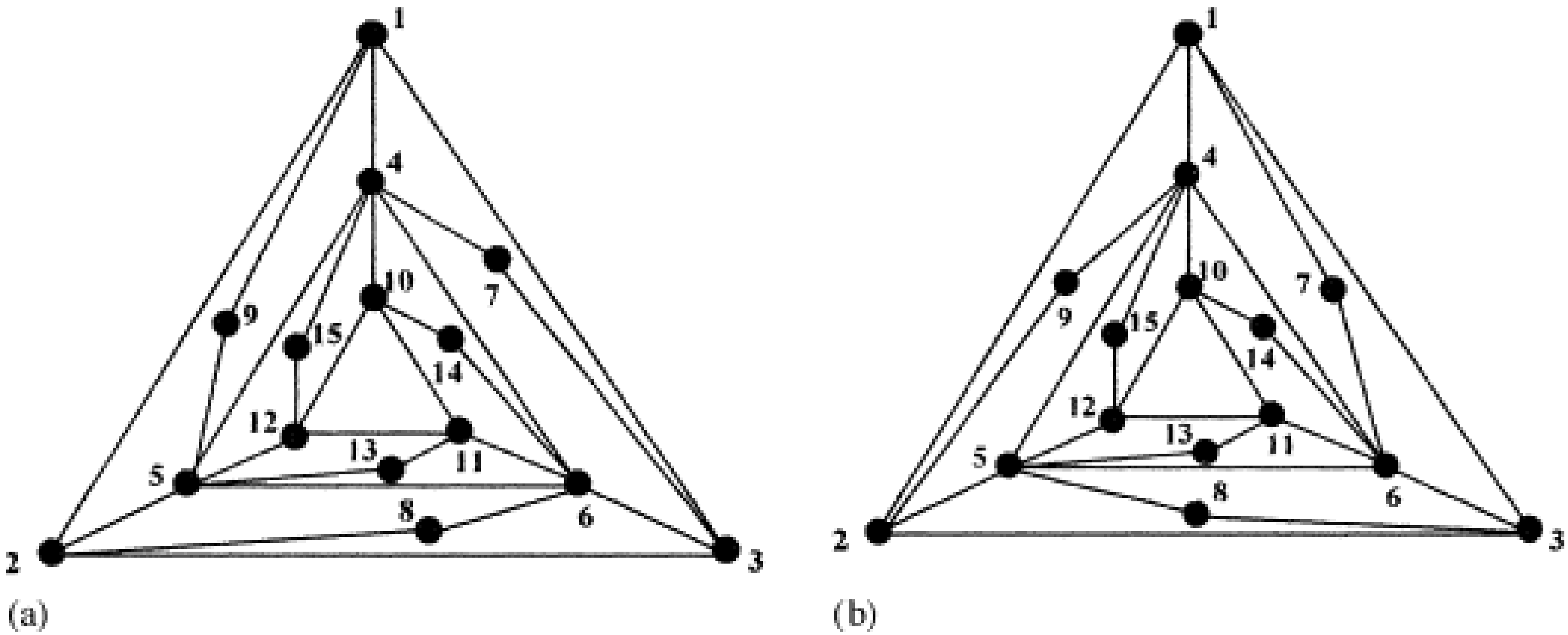
PAL: 6. cvičení

27. 11. 2022

Př. 5/6: izomorfismus

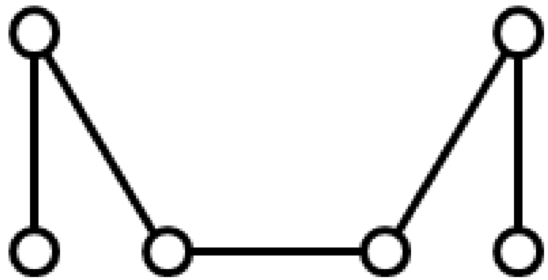
Popište, jak budete co nejefektivněji rozhodovat, zda dva uvedené grafy jsou nebo nejsou izomorfní.



Certifikáty stromů

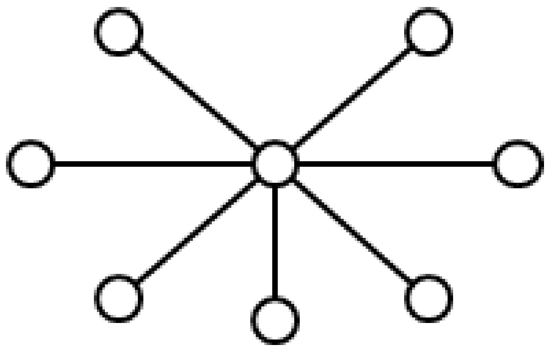
Př. 5/7a: tvorba certifikátu

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.



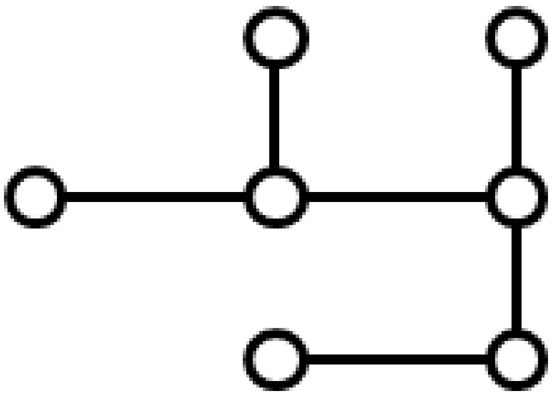
Př. 5/7b: tvorba certifikátu

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.



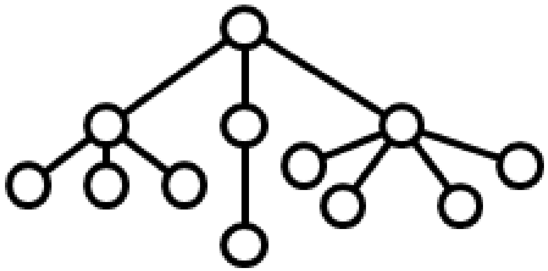
Př. 5/7c: tvorba certifikátu

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. 5/7d*: 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. 5/8: rekonstrukce stromu z certifikátu

Rekonstruujte strom z certifikátu:
a) 0101
b) 0001010110010111
c) 00010110010110010111
d) 00001100010111100100010111111

Př. 5/9: listy v certifikátu

Je dán certifikát stromu. Vysvětlete, jak určíme počet listů tohoto stromu, aniž jej z certifikátu celý rekonstruujeme.

Př. 5/10*: maximální stupeň uzlu z certifikátu

Je dán certifikát stromu. Vysvětlete, jak určíme maximální stupeň uzlu tohoto stromu, aniž strom z certifikátu celý rekonstruujeme.

Př. 5/11*: certifikát stromu s uzly stupně 1 a 3

Popište neformálně, jak bude vypadat certifikát stromu obsahujícího uzly pouze stupně 1 nebo 3. Navrhněte algoritmus, který pomocí certifikátu ověří, zda strom obsahuje uzly pouze stupně 1 nebo 3.

Kombinatorické algoritmy

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 kódu 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 kód G_n.

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 $n \geq 2$.