Zkouska 17.cerven 1999

Priklad 1.: Necht t je vyhledavaci strom s touto vlastnosti:Pro kazdy podstrom t' stromu t plati, ze budto levy podstrom stromu t' obsahuj alespon $\frac{k-1}{3}$ uzlu. Pritom k je pocet uzlu stromu t'.Slozitost vyhledavani v takovem stromu t, vzhledem k celkovemu poctu n vsech uzlu stromu t, je ve tride:
(A) $o(\log n)$ **(B)** $O\left(\frac{n}{(\log n)^3}\right)$ **(C)** $\Theta(n)$ **(D)** $\Theta(\log n)$ **(E)** $O(\sqrt{n})$

Priklad 2.: Kolik existrje vsech AVL stromu na sesti uzlech s polozkami 1,2,3,4,5,6?
(A) 36 **(B)** 4 **(C)** 12 **(D)** 6 **(E)** 1

Priklad 3.: Oznacme:
$f \prec g \Leftrightarrow f \in (g)$
$f \preceq g \Leftrightarrow f \in O(g)$
$f \approx g \Leftrightarrow f \in \Theta(g)$
Pak(povazujeme/li vyraz s n za funkci promenne n) plati:
(A)$\log(n!) \prec \log n^n \prec n \cdot 2^n \preceq n! \preceq n^n$
(B)$\log(n!) \approx \log n^n \preceq n \cdot 2^n \prec n! \approx n^n$
(C)$\log(n!) \prec \log n^n \prec n \cdot 2^n \approx n! \preceq n^n$
(D)$\log(n!) \prec \log n^n \prec n \cdot 2^n \approx n! \prec n^n$
(E)$\log(n!) \approx \log n^n \preceq n \cdot 2^n \prec n! \prec n^n$

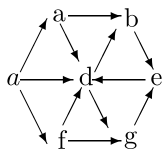
Priklad 4.: Necht a, b jsou realne konstanty.Pak pro funkci 2^{an+b} promenne n plati $2^{an+b} \in \omega(2^n)$, prave kdyz:
(A)$a > 1, b \geq 0$ **(B)**$b > (1-a)n$ **(C)**$a \geq 1$ **(D)**$a \geq 1, b > 0$ **(E)**$a > 1$

Priklad 5.: Strom arity 5 na 31 uzlech, ktery ma hloubku 2, prevedeme na binarni strom metodou "nejstarsi syn/mladsi bratr".Pocet listu(tj.uzlu bez nasledniku) takto vznikleho binarniho stromu je:
(A) 16 **(B)** 10 **(C)** 11 **(D)** 5 **(E)** 25

Priklad 6.: Fibonacciho strom sesteho radu je ohodnocen jako AVL strom cisly 1,. . .,20. Odebereme polozku 2.Kolik lokalnich rotaci se musi provest, aby se obnovila AVL vyvazenost?
(A) 0 **(B)** 4 **(C)**1 **(D)**2 **(E)**3

Priklad 7.: Kazde (sekvencni) asociativni tridici algoritmus ma casovou slozitost ve tride
(A) $O(n \cdot n!)$ **(B)** $O(n!)$ **(C)** $O(n \log n)$ **(D)** $\omega(n/\log n)$ **(E)** $\Theta(n^2)$

Priklad 8.:



Ktere z nasledujich poradi uzlu grafu na obrazku muze odpovidat prohledavani do sirky od uzlu c?
(A) c, d, f, a, g, b, e **(B)**c, a, d, b, f, g, e **(C)**c, a, d, f, b, e, g **(D)**c, d, b, e, g, a, f
(E)c, f, d, a, b, e, g

Priklad 9.: Dijkstruv algoritmus hleda vzdalenosti od daneho uzlu s orientovaneho grafu na n uzlech:

`dijkstra(s,D)=`
{`var S: Set Int; for` $u \leftarrow$`[1..n]do` D_u`:=`∞
$D_s := 0; S := \emptyset;$
$while\ \ S \neq V do\{u := uzel, pro\ \ nejz\ \ D_u = min D_v | v \in V - S;$
$S := S \cup \{u\};$
$for\ \ vsechny\ \ nasledniky\ \ v\ \ uzlu\ \ u\ \ do$
$if\ \ D_u + h_E(u, v) < D_v\ \ then\{D_v := D_u + h + E(u, v)$
$\}\}\}$

```
Modifikujte tento algoritmus tak, aby pro kazdy uzel grafu spocital
nejen jeho vzdalenost od pocatecniho uzlu s,ale i informaci, z niz
lze uricit odpovidajici minimalni cestu.Pouzijte k tomu novy parametr-vektor
predchudcu P delky n, ktery je na pocatku inicializovan nulami.
```