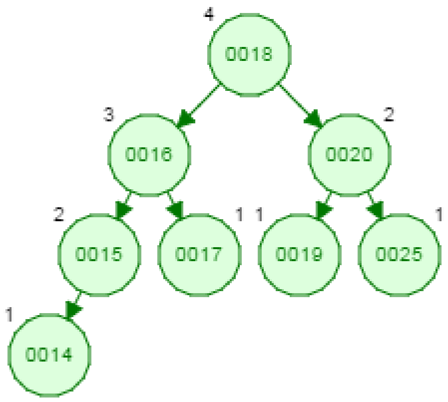
Konstrukcija i analiza algoritama 2, ispit, februar 2015. VREME IZRADE: 150 min

1. Odrediti izgled **AVL stabla** dobijenog izvršavanjem narednog niza operacija nad praznim stablom: (umetni,15), (umetni,11), (umetni,18),(umetni,19),(umetni,25),(umetni,17),(umetni,16),(obrisi,11),(umetni,20),(umetni,14).

Resenje:



2. Vlada je prodajom kompanije XYZ dobila milijardu dolara i spremna je da sav dobijeni novac uloži u vraćanje dugova i multilateralnu kompenzaciju koja je u pripremi. Vlada duguje PIO fondu 1 milijardu dolara, RTB BOR-u 0.2, komercijalnim bankama 5, EPS-u 0.1 milijardi dolara. RTB BOR duguje PIO fondu 0.3, ŽTP-u 0.3, komercijalnim bankama 0.5 i NIS-u 0.3 milijardi dolara. PIO fond duguje NIS-u 0.1; ŽTP duguje EPS-u 0.1 i NIS-u 0.2 milijarde dolara. Komercijalne banke duguju EPS-u 0.2, a EPS NIS-u 0.5 milijardi dolara. Koliki je najveći pozitivni efekat (novčani priliv) koji NIS može da očekuje od multilateralne kompenzacije u koju bi bili uključeni pobrojani pravni subjekti?

Resenje:

Maksimalni novčani priliv koji NIS može da očekuje od multilateralne kompenzacije u koju bi bili uključeni PIO, ŽTP, RTB, EPS je: 0.1 (PIO) + 0(ŽTP) + 0.2(RTB) + 0.3(EPS) = 0.6

3. Konstruisati CRCW algoritam za objedinjavanje dva niza brojeva A i B u jedan sortiran niz. Poredak sortiranja je istovetan kod sva tri niza. Vremenska složenost treba da bude O(1). Na raspolaganju je neograničeni memorijski prostor i broj procesora.

Resenje:

Ako imamo dovoljno procesora i memorijskog prostora treba da napravimo *merge* od dva sortirana niza $a_1, \ldots, a_n$, $b_1, \ldots, b_m$ u vremenu $O(1)$.

Za ovo rešenje trebaće nam $2 \cdot n \cdot m + n + m$ procesora pri čemu ćemo za svaki element a_i niza a imati $m+1$ procesora $P_{i,j}$ $(j = 0, ..., m)$, a svaki element b_j ce obradjivati $n+1$ procesora $Q_{i,j}$ $(i = 0, ..., n)$. Od nizova $a_1, ..., a_n$ i $b_1, ..., b_m$ napravićemo sortirani niz $c_1, ..., c_{n+m}$.

Procesori koji obradjuju (upisuju) elemente niza a radiće sledeće :

* procesori $P_{i,j}$ za $1 \leq i \leq n$, $1 \leq j < m$ rade :
if $(b_j < a_i \leq b_{j+1})$ then $c_{i+j} := a_i$ (jer se u nizu c pre a_i nalazi $i-1$ elemenata niza a i j elemenata niza b)

* procesori $P_{i,0}$ za $i = 1, ..., n$ rade :
if $(a_i \leq b_1)$ then $c_i := a_i$

* procesori $P_{i,m}$ za $i = 1, ..., n$ rade :
if $(b_m < a_i)$ then $c_{m+i} := a_i$

Procesori koji obradjuju (upisuju) elemente niza b radiće sledeće :

* procesori $Q_{i,j}$ za $1 \leq i < n$, $1 \leq j \leq m$ rade :
if $(a_i \leq b_j < a_{i+1})$ then $c_{i+j} := b_j$

* procesori $Q_{0,j}$ za $j = 1, ..., m$ rade :
if $(b_j < a_1)$ then $c_j := b_j$

* procesori $Q_{n,j}$ za $j := 1, ..., m$ rade :
if $(a_n \leq b_j)$ then $c_{n+j} := b_j$

Resenje zadatka se zasniva na činjenici da svaki realan broj pripada tacno jednom od intervala

$$(-\infty, a_1), [a_1, a_2), [a_2, a_3)...[a_{n-1}, a_n), [a_n, +\infty)$$

Pošto svaki procesor radi samo jedan korak vreme izvršavanja ovog CRCW algoritma je zaista O(1).

Primetimo sledeće: primenom algoritma u jednom trenutku više procesora čita sadržaj iste lokacije, ali samo jedan procesor upisuje u jednu lokaciju u jednom trenutku, pa bi opisani algoritam bio primenljiv i u CREW modelu. Takodje primetimo da algoritam radi ispravno i ako ne važi uslov da u objedinjenom nizu nema jednakih elemenata. U slučaju pojavljivanja istih elemenata u nizovima a i b, u niz c se na niže pozicije upisuju elementi niza a.

4. Ako je dat algoritam za množenje dve n * n donje trougaone matrice čije vreme izvrsavanja je O(T(n)), dokazati da postoji algoritam za množenje dve proizvoljne n *n matrice čije vreme izvršavanja je $O(T(n)+n^2)$.
(Može se pretpostaviti da je T(cn)=O(T(n)) za svaku konstantu c)

REŠENJE: Neka su A i B dve proizvoljne kvadratne matrice reda n .
Svaka od njih se moze predstaviti kao zbir po jedne gornje i po jednedonje trougaone matrice(T_A, B_A, T_B, B_B, redom):

$A = T_A + B_A$

$B = T_B + B_B$

Dalje je : $AB = (T_A + B_A)(T_B + B_B) = T_A T_B + B_A B_B + B_A T_B + T_A B_B$

Ako se upotrebi algoritam iz formulacije zadataka moguce je izracunati proizvod :

$$\begin{bmatrix} B_A & 0 \\ T_A & B_A \end{bmatrix} \times \begin{bmatrix} B_B & 0 \\ T_B & B_B \end{bmatrix} = \begin{bmatrix} B_A B_B & 0 \\ T_A B_B + B_A T_B & B_A B_B \end{bmatrix}$$

Kao rezultat dobija se matrica koja sadrzi blokove: $B_A B_B$, $T_A T_B$, $B_A T_B + T_A B_B$

Ovi blokovi ucestvuju u izracunavanju proizvoda $A \times B$.
Matrice $(T_A)^T$ i $(T_B)^T$ su donje trougaone matrice, te se upotrebom datog algoritma moze izracunati blok $T_A T_B$ na sledeci nacin:

$T_A T_B = ((T_B)^T (T_A)^T)^T$
Ukupno vreme izvršavanja je $O(T(2n)+n^2) = O(T(n)+n^2)$.
Na taj nacin se problem izracunavanja proizvoda proizvoljne dve matrice svodi na problem izracunavanja proizvoda dve kvadratne donje trougaone matrice.

5. Ustanoviti da li su sledeća tvrđenja tačna? Obrazložiti netačna tvrđenja primerom ili tačnim tvrđenjem.
a) Nedostatak skip liste je što u najgorem slučaju pretraživanje elementa može da ima vremensku složenost $O(n^2)$.
b) Model CREW ne dozvoljava da dva procesora istovremeno pristupaju istoj memorijskoj lokaciji (u smislu čitanja i pisanja).
c) U neusmerenom grafu sa 2n čvorova nije moguće naći optimalno uparivanje ako je stepen svakog čvora ne manji od n.
d) Svako optimalno uparivanje je i maksimalno uparivanje.
e) U Graham (Grejemovom) algoritmu za konstrukciju konveksnog omotača, tačke se smeštaju u red po FIFO principu.
Resenje:
a) Ne. Kod skip liste u najgorem slučaju pretraživanje elementa može da ima vremensku složenost O(n).
b) Ne. Model CREW ne dozvoljava da dva procesora istovremeno pristupaju istoj memorijskoj lokaciji (samo u smislu pisanja).
c) Ne. U neusmerenom grafu sa 2n čvorova moguće je naći savršeno uparivanje ako je stepen svakog čvora ne manji od n. (po tvrđenju 6.9.1 iz udžbenika)
d) Da. Svako optimalno uparivanje je i maksimalno uparivanje.
e) Ne, tačke se smeštaju po LIFO principu.