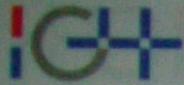


25.03.2023.



CRCW PRAM - učinui razdjelavanja istovremenog pisanja

- 1) prioritetsko - samo procesor s najmanjim indeksom upisuje
upisati
- 2) operacijski - nad svim podacima (koji se upisuju) provodi se
zadane operacija (učvršćuje odabirajući) i upisuje se rez.
- 3) slučajno - jedan slučajno odabran procesor uspijeva upisati
(nedeterministički)

→ default : 3.

Napisati algor. za CRCW PRAM koji će za zadato polje P
proveriti ima li u polju elementata jednaku vrijednost. Za polje od
n elemenata dostupno je u procesora. Rezultat mora biti zapisan
u jednoj varijabli. Bio je?

$$P = [1 \ 3 \ 2 \ 3 \ 4]$$

→ u.p. svih red. $\Rightarrow r_j = 0$, iako je $r_j = 1$

svakci usporeduje je li trenutni el. uiza jednaki vrijesnom



a) $\text{ref} = 0;$

za $i = n \text{ do } n-1$

jednici procesora

parallelno ($\exists j : i+1 \text{ do } n$)

ako ($p[j] = p[i]$)

$\text{ref} = 1;$ // piše se ista vrijednost pa ih može pisati više istovremeno

- br. koraka : $n-1$

- složenost : $O(n)$

b) svaki procesor gleda svoje susjede, ali na raz. udaljenosti :

$$\begin{bmatrix} 1 & 3 & 2 & 3 & 4 \end{bmatrix} \rightarrow I_1$$

$$\begin{bmatrix} 1 & 3 & 2 & 3 & 4 \end{bmatrix} \rightarrow I_2$$

$\text{ref} = 0;$

for ($d = 1 \text{ do } n/2 + 1$)

parallelno ($\exists j : 1 \text{ do } n$)

ako ($p[j] = p[j+d]$)

$\text{ref} = 1;$

- br. koraka : $n/2 + 1$

- složenost : $O(n)$

Isti problem riješit za EREW PRAM na dostupnost procedure reduciraju za proizvoljni binarni operatori

→ učinko ne može čitati i pisati na isto mjesto istovremeno !!
 uprave te kopije :)

sviaki piše rez. na jednu poziciju polja → oduzimaju
 koji reduciraju

parallelno ($\forall a \ j=1 \text{ do } n$)

$$re[0] = 0;$$

$$k[j] = p[j];$$

$\forall a \ d=1 \text{ do } n/2+1$

parallelno ($\forall a \ j=1 \text{ do } n$)

$$\text{ako } (p[j] == k[j+d])$$

$$re[j] = 1;$$

$$re = \text{or_reduce} (re[]);$$

- Vrijednost: $O(n)$

Napisati alg. za CRCW PRAM koji će za polje p odrediti broj razl. elemenata polja.

$$p = [1 \ 2 \ 1 \ 3 \ 4 \ 2 \ 5 \ 1] \rightarrow \text{rg} = 5$$

→ gledam elemente jedan po jedan i uspoređujem ih sa svim, ako postoji isti - ne broji se ; kada dođemo do tog drugog el., ita ujego mnogo više istih - broji se \Rightarrow ukupno se broji samo jednom

$\text{req} = 1$; \rightarrow neprazno polje !

$\exists i \ (i=1 \text{ do } n-1)$

$\text{ink} = 1$; \rightarrow pretpostavi se da će se brojati \rightarrow provjeri paralelno ($\exists j \ (j=i+1 \text{ do } n)$)

ako ($p[i] = p[j]$)

$\text{ink} = 0$;

$\text{rg} += \text{ink}$;

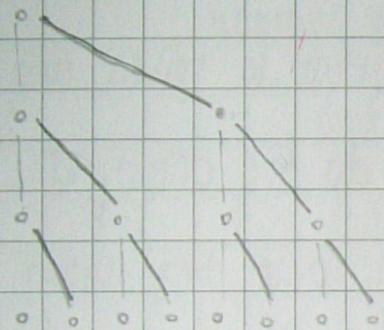
- složenost : $O(n)$

4.

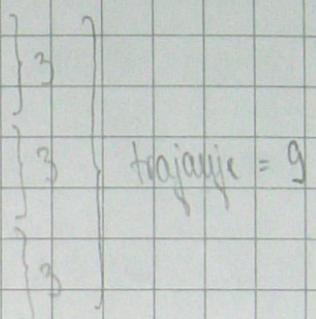
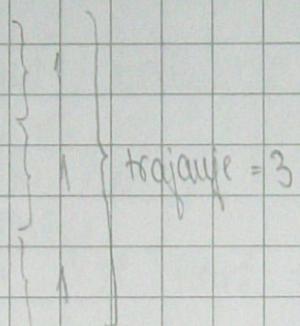
Pr.

Raspodjela vjerojatnosti

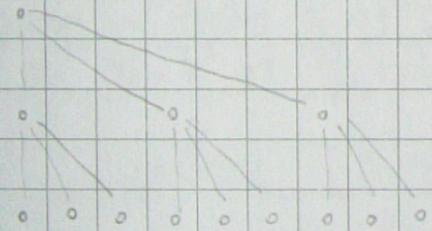
a) binarno stablo



PRAM

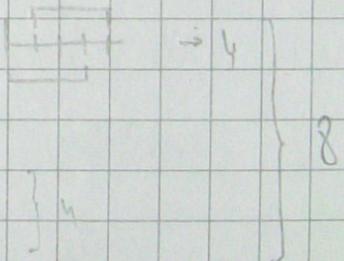
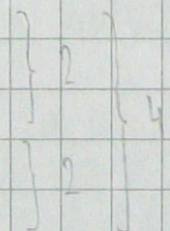
APRAM $d=3$ 

b) terarno stablo



PRAM

APRAM



U APRAM na n procesora u lokalnoj mem. svakog procesora
 učitati se podatku VAR. Napisi program koji će vrijeđnost podatka svakog
 procesora postaviti u vrijednost učitve slijediću podatku

ID - indeks procesora

→ ostalo po defaultu lokalne var

GP [] - globalno polje

(→ svaki procesor rade isti kod :)

GP [ID] = VAR;

ograda

d = 1; → projektor na određenu udaljenost

dok (d < n)

ako (VAR < GP [ID+d])

→ ako je mala vrij. manja od

VAR = GP [ID+d];

vrijednosti, vrijednost upisujem u

svojim

d++;

→ ** ili ako (var > GP [ID+d])

GP [ID+d] = VAR

Ograda

VAR = GP [ID]