

Subiecte Examen APD

Valentin Cristea

ianuarie 2011

331CB:

V1

1. stabilire topologie
 - 1.1 prezentare
 - 1.2. ambii algoritmi pseudocod + explicatii
 - 1.3. de executat pe un graf

2. o problema cu etichete dar schimbata cumva urat...

3. algoritmi genetici
sau
3. modelul foster (cel pe care e facut algoritmul floyd)

V2

1. problema generalilor bizantini
 - descrierea problememei
 - de toata: conditii de consistenta interactiva
 - solutia cu mesaje scrise: conditii, descriere, pseudocod, teorema care dem corectitudinea
 - un exemplu practic

2. o problema cu etichetare zonelor, banuiesc ca ai auzit de ea

3. 3.1 sa scrie suma prefix atat ca SIMD cat si ca MIMD
3.2 problema terminarii
 - descrierea problemei
 - algoritmul Dijkstra - Scholten
 - complexitate

V3

1. problema generalilor bizantini
 - descriere, consistenta interactia, solutia cu mesaje orale, un exemplu practic

2. sa explicam daca merge si in cazul in care nu merge, cum am modifica algoritmul de stabilire a topologiei sondaj-ecou pentru un graf care are topologie

dinamica (unele noduri pot sa pice si pot sa revina)

3. 3.1 producator consumator cu buffer limitat, cazul simplu si cazul cu mai multi producatori + consumatori (parca asa suna)

3.2 Lelann-Chang-Roberts

V3

1. Detectia terminarii in procesele distribuite

1. Enuntarea problemei. Algoritmul de terminare pentru o topologie inel.

2. Algoritmi Dijkstra-Scholten si Huang. Descriere. Pseudocod.

3. Fie algoritmul de detectie terminare, folosind culori si jetoane: (se da un pseudocod pentru un proces...).

Sa se determine daca algoritmul este corect. Daca da, sa se determine si sa se explice conditia de terminare.

Daca nu, sa se justifice de ce si sa se spuna ce imbunatatiri se pot aduce pentru a face algoritmul corect.

2. Consideram o topologie dinamica de procese, in care dorim stabilirea topologiei.

Nodurile pot "pica" in timpul algoritmului de aflare topologie. Un nod picat face ca toate legaturile sale sa pice, iar un mesaj trimis pe o legatura picata este pierdut fara a se primi avertisment. Sa se determine daca algoritmul clasic cu sondaj-ecou da rezultate corecte pentru o astfel de topologie dinamica. Daca da, sa se scrie algoritmul sau partea relevanta din acesta care face ca rezultatele obtinute sa fie corecte si in acest caz. Daca nu, sa se stabileasca un algoritm imbunatatit de aflare a topologiei.

3. 1. Cautarea paralela

a) Descrierea problemei

b) Algoritm (descriere + pseudocod)

c) Complexitate

(sau) 2. Algoritmul pulsatiilor

a) Varianta pentru un diametru D al retelei; b) Varianta pentru o retea conexa.

334/335CA

V1

1.

Algoritmi unda: arbore

a) Explicatia alegerii algoritmului unda. Descrierea algoritmului.
(0.5p)

b) Descrierea algoritmului in pseudocod. Justificarea corectitudinii.
Complexitate. (1.1p)

c) Era un desen langa enunt. Se cerea aplicarea algoritmului pe acel desen (desen identic cu cel din notitele de curs). (0.4p)

2. Etichetarea regiunilor

Se da un tablout[1:n,1:n] cu valoarea inte pixelului respectiv. Sa se descrie un algoritm heartbeat care eticheteaza regiunile cu intensitati egale.

Descriere = 0.3p

Pseudocod = 0.7p

3.1 Modelul work-depth

- nu am retinut defalcarea

3.2 Hirschberg-Sinclair:

- Descriere 0.3 p

- Pseudocod 0.4p

- Complexitate 0.3p'

V2

1.1. Problema producatori-consumatori. Descriere.

1.2. De explicat primitivele P si V cu notatiile de atomicitate si sincronizare.

1.3. Pseudocod cu comentarii

2. De aflat un arbore de acoperire fara a afla mai intai topologia (un fel de sonda-ecou zic eu...) .

3. 1. Complexitate: numar de mesaje. Descriere si aplicatie pe algoritmul pulsatiilor.

la alegere cu:

3.2 Problema produs prefix ,implementare SIMD si MIMD si de explicat diferentele intre ele.

V3.

1. Comunicari sincrone si asincrone prin mesaje

a) Prezentarea generala a problemei. Cum se realizeaza partajarea datelor? Cum se realizeaza sincronizarea? (0.5p)

b) Constructii pseudocod pt transmitere sincrona si asincrona. Exemple (1p)

c) Cum se reprezinta transmiterea sincrona si asincrona a mesajului in MPI. (0.5p)

2. Se dau doua seturi S si T cu valori intregi. La fiecare pas, se interschimba o valoare din S cu una din T. Descrieti (0.3p) si scrieti (0.7p) algoritmul astfel incat, la final, valorile din S sa fie mai mari decat orice valoare din T.

3.1 Algoritmul de detectare a terminarii folosind confirmarile mesajelor - Dijkstra-Scholten.

a) Descrierea problemei (0.3p)

b) Pseudocod (0.4p)

c) Complexitate (0.3p)

3.2 Algoritmul lui Finn.

a) Descrierea algoritmului (0.3p)

b) Pseudocod (0.4p)

c) Complexitate (0.3p)

V4

1. Problema generalilor bizantini

a) Prezentarea generala a problemei (0.5 p)

b) Prezentarea solutiei ce foloseste mesaje orale (aici cerea cam tot : teoreme, pseudocod, demonstratie ca algoritmul functioneaza, etc. 1.2 p)

c) De reprezentat grafic problema generalilor pentru 3 generali (unul din locotenenti, la alegere, era neloial) (0.3 p)

2. Se dau n numere si o secventa de n procesoare organizate in linie de asamblare (de fapt se dorea o structura vectoriala). Prezenta functionarea algoritmului: P[0] are toate cele n numere, pastreaza valoarea cea mai mica si trimite mai departe restul numerelor lui P[1]. Procesul se repeta pana cand la procesorul P[n]

ajungea ultima variabila (cea mai mica valoarea).

Cerinte :

2.1 Prezentarea solutiei (care, paradoxal, era prezenta in enunt :D - mi s-a spus sa fac un desen, sa detaliez putin pe pasi) (0.5 sau 0.3 p)

2.2 Implementarea algoritmului in limbaj pseudocod

3.1 Complexitatea algoritmilor paraleli.

a) Metrici

b) Exemplificare pe sortarea unui vector, folosind o structura de N procesoare

3.2 Huang

Explicarea mesajelor, a modului de functionare, pseudocod (Nu am citit cu mare atentie)

331CA

V1

1. Alg prefix -> lista inlantuita cu valorile end[i] setate initial la leg[i] (unde leg[i] e urmatorul din lista dupa i). La sf end[i] trebuia sa contina referinta catre ultimul element.

2. -Pseudocod + Analiza complexitate ptr problema urmatoare din curs (din cursuri de ceasuri logice, a 5-a pagina) :

Un proces care dorește să intre în secțiunea critică trimite mesaje de cerere request tuturor celorlalte procese. Pentru a putea intra efectiv în secțiunea critică, este necesar

să primească de la fiecare câte un mesaj de răspuns reply. La recepția unui mesaj request, un proces poate determina dacă el sau procesul care a facut cererea ar trebui

să intre în secțiunea critică. Când el are prioritate, mesajul reply este întârziat; altfel,

el este transmis imediat procesului ce a generat cererea.

3. La alegere intre

a) modelul Foster. Replicated workers cu exemplificare pe Floyd

b) terminarea in inel prezentare + pseudocod + complexitate

V2

1. Stabilirea topologiei. Mesaje de sondaj cu ecou. Enuntarea

problemei, pseudocod, analiza complexitatii. Mentionati si alte metode de stabilire a topologiei.

2. Avem $n*n$ procese dispuse pe un grid periodic de numere intregi. Fiecare casuta are vecini la nord, sud, est, vest. Pseudocodul algoritmului de aflare a maximului din matrice, astfel incat in final, toate casutele din matrice sa aibe valoarea maxima. Complexitate.

3. La alegere intre:

3.1. Calculul prefix. Pseudocod, analiza complexitatii.

3.2. Algoritmi genetici... something...(nici nu l-am citit pana la capat)

332/333CA

V1

1. Atomicitate si Sincronizare. De ce sunt importante pentru algoritmi paraleli care folosesc variabile partajate...sau ceva de genul. Cum se scrie in pseudocod si semantica. Exemplu de folosire cu P si V. Cum se evidentiaza in java atomicitatea? Dar sincronizarea?

2. O problema in care se considera un tablou de procese. Fiecare proces are o valoare intreaga si o trimite celorlalte procese. Cand termina de trimis, se termina si procesul. Ultimul proces ramas detine valoarea minima din tabloul initial. (0,3 p) ideea. (0,7) pseudocod

3. 3. 1. Alegerea liderului intr-o topologie inel, in care se trimite doar in sensul acelor de ceasornic si pot fi mai multi initiatori

3. 2 . Stabilirea topologiei cu sondaj. (cred)

V2

1. Alegerea liderului. De ce sunt potriviti algoritmi unda. Idee,pseudocod. Alte metode de alegere a liderului

2. Se da o matrice de N^2 procese. Fiecare proces detine cate un vector de P elemente. Se cere suma elementelor cu acelasi index de la toate procesele.

3. La alegere intre:

- Producator cosumator . varianta cu mai multi producatori si un consumator, mai mult consumatori si un producator. Diferentele dpdv al sincronizarii

- Algoritmul de terminare intr-o topologie care admite un ciclu de lungime nc care trece prin toate nodurile. Pseudocod + idee

V3

1. Ceasuri Logice 1.1 De ce ? 1.2 Cum sunt implementate ? 1.3 Desen cu 3

procese si schimb de mesaje intre ele: de specificat ceasul fiecarui eveniment 1.4

Descrierea algoritmului de excludere mutuala cu semafor distribuit 1.5

Pseudocodul de la 1.4

2. 3 procese paralele detin fiecare cate o secventa ordonata de numere. Cele trei secvente au cel putin un numar comun. Sa se determine acest element comun minim. 1.1 Ideea 1.2 Pseudocod

3.1. Algoritmul Heartbeat 3.1.1 Idee 3.1.2 Pseudocod

3.2 Cititori Scriitori : idee, pseudocod. Pot apare situatii in care accesul scriitorilor este blocat de cititori? Explicati.

Ideea de baza la 2 era aceea ca procesele isi transmiteau primul element din secventa (elementul minim). se facea maximumul intre acestea si fiecare proces stergea primele elemente din secventa sa mai mici decat acesta (fiindca nu aveau cum sa se regaseasca in celelalte secvente ale celorlalte procese). Se repeta acest pas pana cand se ajungea ca elementul comun minim sa fie primul element in fiecare secventa. Poate ca exista si alte implementari, dar aceasta solutie a fost punctata la minim 2 persoane.