3 Paralelni arch

1. [1b] Common CRCW a Random CRCW. Ktery je silnejsi a proc?
zduvodneni je nekde ve starsich zadanich pripadne ve skriptech u popisu CRCW

2. [2b] Common CRCW a Random CRCW. Ktery je silnejsi a proc?

3. [4b] popiste a vysvetlete polylogaritmicky slozitou simulaci jednoho kroku Prioritni CRCW-PRAM(n,p) algoritmu na EREW-PRAM(n+O(p),p) pocitaci. Predpokladejte jednotkovy casovy model, R,L,W=1

4. [4b] Polylogaritmicka simulace prioriti CREW na EREW.

5. [4b] Odvodte pocet procesoru p' a casovou slozitost t' **simulace** PRAM(n,p) algoritmu s casem t = T(n,p) na PRAM (m,p') pocitaci tehoz typu, je-li m < n. Tuto **simulaci popiste**. Jake dalsi predpoklady jsou treba? Uvazujte **jednotkovy** casovy model, kdy operace R,W a L trvaji cas 1.

---

a. Urcit, ktery model CRCW PRAM je vypocetne silnejsi, zda Prioritni(Priority) ci Nahodny(Arbitrary).

Na toto jsem si nedokazal odpovedet, ani nikdo, koho jsem se ptal, az na kolegu Tvrdika... ;-)

Plati, ze Prioritni CRCW >= Nahodny CRCW, to vime vsichni, ted proc. Vysvetleni:
mam algoritmus pro nahodny crcw, to znamena, ze do bunky pameti zapise nahodne vybrany procesor, z tech, ktere o to maji zajem. Kdyz takovy algoritmus nasadim na prioritni, tak se vybere procesor s nejvetsi prioritou misto nahodneho, ale to je nam u alg. pro nahodny crcw uplne jedno, alg. funguje.

Naopak mame-li alg. pro prioritni crcw, tak pocitame, ze se vybere procesor s nejvetsi prioritou, ale vybere se nahodny, takze algoritmus nemusi nutne fungovat.

Takze zadne algoritmy apod. -- jak jimi nekdo strasi, dnes mi doc. Tvrdik rikal na zkousce prave toto, ze to je spravne reseni ukolu 1. Zadna veda, ale pokud ma clovek malo fantazie, tak to u zkousky nevyplodi.

b. Určit jestli je silnější Prioritní nebo Náhodný CRCW PRAM a dokázat.

Tady chtěl napsat že to je prioritní a dále se muselo napsat že prioritní může simulovat neprioritní a jak se to udělá (!!!to tam chtěl) a že náhodný nemůže simulovat prioritní bez nějkého zpomalení a PROČ.

c. [1b] ktery model je vypocetne silnesi Prioritni nebo Nahodny CRCW PRAM
dokazte

d. polylogaritmickou simulaci prioritniho p-procesoiroveho CRCW PRAM na p'-proc EREW PRAM s pameti (p+m). *Popsat algoritmus (viz skripta, takove to s pomocnym polem A). Kolik bude potreba minimalne procesoru (p') ? (Spravne je p - protoze je to EREW a kazdy musi simulovat jednoho).*
Chtel to docela presne, vyptaval se na kazde slovo, ktere bylo ve skriptech a ja jsem ho v odpovedi vynechal. (4 body)

4 Propojovací sítě

1. [1b] Odvodte vyraz pro prumer n-rozmerneho zabaleneho motylka wBFn.

2. [ ] Prumer CCCn

3. [2b] Dokazte, ze 2-D toroid K(z1,z2) je uzlove symetricky.

4. [1b] Pro zadany uzel $x \in V(Qn)$ odvodte vyraz pro pocet uzlu vee vzdalenosti i od x.

5. [3b] 4d Toroid, zadany dva konkretni body lisici se v souradnicich 3 dimenzi. Vyjadrete algebraicky nejkratsi disjunktni cesty mezi nimi (i 2.nejkratsi).
U toroidu je to jine nez na hyperkrychli!!!!(coz spouste lidi doslo az po zkousce)zalezi , ktere body se spojuji,jestli nahodou neni kratsi cesta okolo
Obecne je tam tolik cest, kolik je stupen uzlu(coz je u tohoto

toroidu 8 cest) Protoze se ty body lisily ve trech dimenzich, tak jsou tam tri nejkratsi cesty, Dale dve, ktere jsou delsi o 2 a zbytek ktere jsou delsi o 4

---

a. [1b] Charakterizujte uzlovou a hranovou symetrii 3D toroidu T(z1, z2, z3), odpoved dokazte.

b. [1b] Dokazte ze v Qn nelze zkonstruovat hamiltonovskou cestu P(u,v) mezi uzly u=1010110 a v=1100011
[1b] Dokažte, že v Q7 nelze zkonstruovat cestu P(u,v) mezi uzly u=1010110 a v=1100011.

c. [1] chybovy prumer na huperrychly,DOKAZTE
[n - protoze dva nejvzdalenejsi uzly maji mezi sebou n disjunktnich cest a ty musime vsechny prerusit]

d. chybovy prumer Qn

e. Uvedte a dokazte vyraz pro chybovy prumer Qn

f. [1] odvodit prumer na wBFn
n+(spodni zaokrouhleni)(n/2)

g. pocet automorfismu Qn - odvodit - *viz skripta, popis Qn*

5 VNOROVANI

1. [3b] Vnoreni M(22,24) do M(x,5) s load=2 a min dil, urcit ecng.

2. [2b] Dokazte, ze lze libovolnou hemitnovskou kruznici s n uzly vnorit do lib.souvisleho n uzloveho grafu s dil <=3.

3. [4b] Odvodit vztah pro minimalni dilataci pri vnoreni D2 toroidu K2(z1,z2) do 1D toroidu K1(z1*z2) pri loadu 1.

4. [3b] CBNn -> Qn+1 - algoritmus s load=ecng=1, dil=2 aplikaci problemu zkonstruujte a algebraicky popiste vnoreni CBN3->Q4

5. [3b] Vnoreni CBTn do Qn+1 a ukayat v realu na CBT3a Q4.

6. [1b] Dokazat, ze CBTn neni podgrafem Qn+1.
[protoze Qn+1 je VYVAZENY bipartitni graf, kdezto CBTn ani nahodou->stane se jim pri zdvojeni korene]

7. [2b] Jake muze byt maximalne y v M(4,y), kterou se snazim nacpat do M(16,20) s load =2, dil=1. Spocitejte take hranovou zatizenost. (y=136,ecng=2)
Reseni:
str.87 - POZOR! vsechny rovne useky hada maji LOAD presne 2! Delam to tak, ze davam vzdy dva uzly do jednoho, tim mam load=2 a dil maximalne 1, pulka hran dokonce zanikne, protoze vedou z uzlu do toho stejneho. Na koncich je klasicke prelozeni ze cviceni - spocital jsem tak, kolik uzlu se vejde do M(16,20), vydelil 4ma a dostal y.

8. [1b] Dokazte, ze lze libovolnou kruznici s n uzly vnorit do lib.souvisleho n uzloveho grafu s dil <=3.

---

a. Vnoření CBTn do Qn+1.
Chtěl napsat že se to dělá zdvojením kořene a jak se to dělá a obrázek pro CBT3. Tohle jsem všechno měl, ale chyběla mi nějaká ta mapovací fce. jak se ta druhá krychle má otočit vůči té první. Takže místo plného počtu 3b jsem měl zase jen 1 :))).

b. [3b] vnoreni CBTn do Qn+1, popsat, ukazte na CBT3

c. [3b] CCCn a wBFn, dokazat, ze jsou kvaziizometricke, nakreslit pro n=3

d. [2b] Max y takove, ze mrizka M(3,y) lze vnorit do mrizky M(7,9) s dil=1 a load=2. Schema reseni.
[2b] Určete max. Y takové, že mřížku M(3,Y) lze vnořit do M(7,9) s dil=1 a load=2.

e. [3b] Dokazte, ze do kazdeho souvisleho grafu G s n uzly lze vnorit kruznice s n uzly s load=1, dil=<3. Urcete jeji ecng.

f. [2]Vnoreni Q2n do T(2^n,2^n) s load=1, jaka je za teto podminky spodni mez a proc!!!
skripta lema 5.2

g. spodni mez pro dilataci pri vnoreni Q2n -> T(2^n,2^n)
Vypocitejte spodni mez pro dilataci vnoreni Q2n do 2D toroidu T(2^n,2^n), uvazujte pouze vnoreni s load=1.
Tady mel nejakou chybu v zadani, uz nevim jakou - nekdo me doplnte..

h. [2b] jak se vnoří M(x,y,z) do Qn s dil=load=1 ukázat na dvou bodech a zjistit jejich vzdálenost v té krychli

i. [2] vnoreni mrizky M(11,11) do M(3,y), pri load=2, minimalni dil, urcit ecng
y=22, dil=2, ecng=4
AAB na mrizce , SF, urcit t_aab, r

6 prefixovy soucet

1. [3b] Preorder na PRAMu. str.110-112. Zadano bylo jen EA >> nutno dopocitat DIR u kazde hrany podle toho, jestli je jeji Eukl. cislo mensi nez u protidvojcete.
Dale na tomto spocitanem poli (1=dopredna,0=zpetna) provedu PPS.
Uzel ma index X z vysledku PPS, prave kdyz do nej vede dopredna hrana.
Dale se melo spocitat T(n,p),C(n,p) a Fi1 (bo co sou ty widle za pismenko), to jsem proste nejak vygeneroval....nevim, jak se to poradne pocita doted.

2. [5] Provest PPS na APRAM! pocitaci. Zase chtel T,C, atd....
APRAM pocitac postupuje stejne jako PRAM pocitac, akorat se musi prokladat barierama, neco jsem tam psal, ale nerad bych kecal, tak to sem radsi uvadet nebudu.
Melo to byt pro p<n procesoru, takze vlastni vypocet slozitosti mel sekvencni a paralelni cast. Ta paralelni pak vypadala (RRLBWB)^logp. Z toho zjistit T(n,p) dosazenim za R,L,W,B, vypocitat C(n,p), Psi1, Psi2.

3. [4b] PPS na EREW PRAM, p<n, L 1, R/W d > 1, Dokazte spravnost! Odvodte T(n,p), E(n,p)

4. [6b] Uvazujte standardni APRAM model pocitace (lokalni operace trvaji cas 1, posloupnost k>=1 globalnich R/W za sebou trva cas k+d-1, kde d>1 je konst., barierova synchronizace p procesoru trva B(p)=d*logp). Dal predpokladejte, ze procesory maji nezavislou pametovou a aritmetickou jednotku. Popiste co nejrychlejsi cenove optimalni APRAM algoritmus pro paralelni redukci n cisel pomoci binarni asociativni operace. Odvodte paralelni cas T(n,p), cenu C(n,p) a izoefektivni fce Psi1(p) a Psi2(N).

5.[4b] PPS na jednorozmernem poli , kde lokalni PPS probiha na Qn s p < n a WH. Zadany casy pro WH (ts + d.td + mi.tm). Algoritmus, T(n,p), a pri E0 spoctete psi1,psi2.
Tady slo o dve veci. Udelat PPS na Q(ten algoritmu se zlutymi a zelenymi registry. Slozitost je stejna jako u binarni redukce A pak to cele aplikovat na pole, tj kazdy procesor dostane n/p hodnot, na kterych si udela PPS a pak posle vysledek do procesoru p+1 a nasledne si vsechny procesory upravy svuj PPS
, ze si k nemu pricctou tu hodnotu od procesoru p-1(je to ve skriptech nekde u PPS)

6. [4b] PPS na Qn s p < n a WH. Zadany casy pro WH (ts + d.td + mi.tm). Algoritmus, T(n,p), a pri E0 spoctete psi1,psi2.

7. Paralelni binarni scitacka - popsat algoritmus

8. [2b] PRAM algoritmus pro nalezeni eulerovy cesty, spocitat cas a tak.
[algoritmus je ve skriptech, zbytech to chce odvodit]

---

a. [4b] PRAM algorimus cislovani uzlu stromu postorder

b. [2b] Popiste paraelni algoritmus pro vypocet Eulerovske cesty ve stromu(s n uzly a p procesory, a p=<n)a jeho reprezentaci(toho stromu) na PRAM pocitaci a jeho slozitost(toho algoritmu).

c. [3]vytvoreni pole rank na zretezenem seznamu na CREW PRAM pocitaci, algoritmus, posudte skalovatelnost
ve skriptech, nevim kde, pokud mate jen algoritmus pocitejte s 1 slovy "jednim" bodem, je to kruty ale je to tak, me dal 1,5 proto ze jsem to mel odvozeno, ale blbe ale mel jsem tam navic naznak toho vylepsenyho - lepe skalovatelneho algoritmu

d. [5] par. redukce na APRAM - cenove optimalni myslim, *to sem nemel a dost me to nakrklo myslel sem ze todle umim* **-** *"DYK JE TO PRECI JASNY" takze to nepodcenujte, samozdrejme abyste dostali vice nez 1 bod musite umet odvodit vsechny ty psi a vubec vsechno kolem toho*

e. popsat algoritmus pro vypocet poradi od konce ve zretezenem seznamu nelikosti n na CREW PRAM s n procerory

f. paralelni redukce na APRAM. T(n,p), C(n,p), psi1, psi2

g. APRAM cenove optimalni par. redukce - stejny jak v materialech na webu (vyp Tn,Cn,psi1,psi2)

h. sudo-licha redukce na hyperkrychli

i. Sudo-licha redukce reseni tridiagonalni soustavy n lin.rovnic mapovanych blokove na p-proc. hyperkrychli p<n. Odvodte asymtoticky Tn. Vseportove smerovace s WH (t(d,mi)=ts+dtd+mitm). Standartni model pocitace - aritm/log/pametove operace maji konstantni slozitost.

j. Napiste casove optimalni alg. paral. vypoctu poradi od konce (rank) prvku v retezenem seznamu o vel. n na EREW PRAM s n proc. Vstupem alg. je zret. seznam ve sdilene pameti PRAM reprezentujici pole nasledniku. Jaka je skalovatelnost alg?

k. [3b] PPS na 1-port 1D M, p<n jak se změní, když bude M 2-port

l. [5] PPS na APRAM, urcit fi1,fi2
fi1=plogp asi, fi2=n/log(n) asi

m. zkonstruujte nejkratsi cestu v CCCn (n=6, zadany hodnoty, jeden mel i=1, druhy i=4, lisily se navic ve 3 dimenzich)
n. PPS na Qn (n=log p). WH prepinani, popsat algoritmus (takovy ten se zelenymi a zlutymi registry), T, fi1, fi2, fi3, Tmin. To vse za 5 bodu.

7 Třídění

1. [4b] Bitonic Merge Sort na Qlogp, T(N,p), psi(p), psi(n), sekvence k cisel - a*k*logk, Merge and split 2 posl. b*k

2. [5b] Shearsort na 2D mrizce(sqrt(p),sqrt(p)), N cisel 1<=p<=N, spocitat T, fi1, fi2, fi3, posoudit skalovatelnost a dokazat ze shearsort funguje.
[cele ve skriptech, dukaz pomoci 0-1 vety]

3. [5b] Popiste algoritmus pro **trideni** N celych cisel na p-procesorove **3-D mrizce** M($\sqrt[3]{p}$, $\sqrt[3]{p}$, $\sqrt[3]{p}$), p <= N. **Odvodte** co nejpresneji jeho paralelni cas T(N,p) a **izoefektivni** funkce ψ_1(p) a ψ_2(p), jestlize (1) **sekvencni setrideni** k cisel na 1 procesoru trva cas αklogk, (2) operace **Merge-and-Split** 2 setridenych posloupnosti k cisel provedeny 2 sousednimi procesory trva cas βk a (3) pro trideni v 2-D mrizkach je pouzit algoritmus Shear-Sort.

4. [4b] Transpozice matice na M(sqrt(p),sqrt(p)), WH, XY, vseport ...

---

a. Shear-Sort, slozitost, psi1, psi2, psi3, popsat algoritmus.

Kdyz algoritmum moc nerozumite a jen si pamatujete casovou slozitost, muzete zle narazit. Chtel vedet, proc se provadi logn krat trideni sloupcu a radku atd. Takze dukaz podle vety 0-1 si radeji ctete. ;-) Tedy alespon mne se to poradne vyplatilo.

b. Shear-Sort na 2-D mřížce.
Klasika, chtěl algoritmus, paralelní čas a efektivitu a všechny fí. Pokud jste měli jen popsaný alg. a !!!dobře par. čas tak to bylo opět jen za 1 bod, jak jinak.

c. [5b] shear-sort 2D-mrizka M(sqrt(p),sqrt(p)) vypoctete T, psi[1-3], slovne   zhodnotit skalovatelnost

d. [6] BMS pro trideni N cisel po radcich na 2D mrizce M(sgrt(p),sqrt(p)), 1<=p<=N a sqrt(p) mocnina dvou. Spocitat T(N,p), Fi1 a Fi2. Vseportova s plne duplex. linkam a prepinanim SF. Sekv. trideni k cisel na 1 proc. trva (alfa)(k)log(k) a merge sort dvou setr. posloupnosti k cisel trva (beta)(k)d, alfa a beta jsou konstanty.
[] Popište Bitonic sort N čísel po řádcích na 2D M(sqrt(p),sqrt(p)) p<=N a sqrt(p)je mocnina 2. Určete T, psi1, psi2, jestliže je mřížka všeportová, full duplex, SF. Sekv. třídění k čísel na 1 proc. trvá alfa*k*LOG(k) a MS dvou posloupností k čísel trvá beta*k*d, d- vzdálenost procesorů.

e. [6b]BitonikSort asi na 2D mrizce...to uz nevim, to sem nestihal udelat(a ani neumel).

f. [6b] 3D sort, algoritmus, důkaz pomocí 0-1, T, psi1, psi2

8 směrování

1. [4b] jednoduchy neorientovany souvisly graf G s N uzly. Popiste algoritmus konstrukce smerovaci funkce odolne vuci zablokovani na prime propojovaci siti s topologii grafu G s plne-duplexnimi kanaly a vse-portovymi smerovaci s uzly oznacenymi 1..N. Konstrukci demonstrujte na prikladu

```
    2-3
   /\ /\
  4--1--5
 / \/ \/ \
 6--7--8--9
 \ / \/ \ /
 10--11--12
```

2. [4b] Nejaky graf a udelat pres kostru bezkolizni smerovani - netusim

3. [5b] 2D toroid, vseportovy OAB, urcit spodni mez poctu kroku ro=? a cas.narocnost tau=?, navrhnout algoritmus, spocitat jeho r(pocet kroku) a t(cas), srovnat s idealem.
[tady to chce tu diagonalu, viz skripta]

4. [4b] Jednoduchy neorientovany souvisly graf G s N uzlama. Algoritmus konstrukce smerovaci fce odolne vuci zablokovani na prime propojene siti s topologii grafu G s fullduplex kanaly a vseport smerovaci a s uzly 1-N. Konstrukci jeste demonstrovat na zadanym grafu. Ten sem kreslit nebudu - kdyztak at me nekdo doplni...
Mela se vymyslet smerovaci funkce pro zadany graf, ktera by zajistila bezkolizni smerovani. (Zrejme chtel to co je na konci 8. kapitoly.)

---

a. Dokázat tu bezkoliznost na Qn.
No tady jsem vyšel z motýlka a napsal že směrování za předpokladu dané permutace na něm je bezkolizni a potom že krychle se chová podobně jako tento motýlek (Q2 je vlastně elementární motýlek),...
Nejdřív mi za to dal plnej počet, ale pak do toho začal šťourat a ptal se ještě na nějaký dodatky no a nějak jsem to zmotal a on prohlásil "Pane kolego vždyť vy tomu nerozumíte." a dal mi jen jeden bod a ještě dodal "Správně by to mělo být za nula.".

b. spodni mez pro Multicast na mrizce, algortimnus a ukazat na M(5,5)

c. 4. 2D mrizka M(Zx,Zy) s plneduplexnimi linkami a 1-portovymi procesory, jejichz smerovace jsou WH. Smerovani XY. Odvodte spodni mez na pocet kroku pro multicas se zdrojem v s<>(0,0) a s cilovou skupinou uzlu S a pak obecne popiste krokove optimalni alg. pro tento problem. Aplikujte ho na M(5,5) - s je 2, S jsou 1, ostatni 0:

```
0 1 1 1 0
1 1 1 1 0
0 1 0 0 1
1 2 1 1 0
1 0 1 1 0
```

9 Permutace

1. [4 b.] Odvodte casovou slozitost permutace otoceni pir: un-1...u0 |-> u0...un-1 na vseportovem SF neprimem motylku indBFn (viz. obr. pro n=4 - byl nakresleny v zadani), je-li pouzito minimalni on-line smerovani, cili pro vsechna i z {0,1}n paket ze vstupu i je poslan nejkratsi cestou na vystup pir(i).

2. [3b] Minimalni smerovani na motylku, permutace otoceni, odvodit casovou slozitost.
[odmocnina(N) viz. skripta]

3. Kolik permutaci udela sit Omega

4. [3b]Uvazujte **neprimou** sit **obycejny motylek** oBFn s 2^n vstupy a 2^n vystupy. Prepinani paaketu je typu uloz-posli-dal (SF). Necht $1 < m < 2^n$ a A={a1,a2,...,am} je usporadana podmnozina mnoziny vstupu, kde $0 <= a1 < a2 < ... < am <= 2^n – 1$, a

B={b1,b2,...,bm} je usporadana podmnozina mnoziny vstupu, kde 0 <= b1 < b2 < ... < bm <= 2^n – 1. Pak zobrazeni π: A -> B takove, ze $\forall$i: π(ai)=bi, nazveme **monotonni permutaci** A na B. Dokazte, ze kazdou monotonni permutaci lze na neprimem SF oBFn realizovat **bezkolizne** a proto v O(n) krocich. Odhadnete skrytou konstantu. /3b/

---

a. [3b] oBFn, SF, monotonni permutace. Dokazat, ze kazdou monotonni permutaci lze na SF oBFn realizovat bezkolizne, tj. v O(n) krocich, odhadnete skrytou konstantu

b. [1b]kolik udělá permutací neprímá síť, k úrovní, n čísel na vstupu, jeden přepínač 2x2

c. [4] proc je kolizni permutace prohozeni na motylku+ jak to bude rychlejsi?
horizontalne se rozpuli motylek a vsechny vstupy se prehodi z 1 motylku na vystup druheho a obracene; reseni obrazkem (trivialni prej rika kolega Hudy :))

10 Kolektivni komunikacni algoritmy

1. [3 b.] Odvodte spodni meze RoAAB(Qn,m) na pocet kroku a TauAAB(Qn,m) na casovou slozitost vysilani vsichni-vsem (AAB) v 1-portove plne duplexni kombinujici hyperkrychli Qn s prepinanim uloz-posli dal (SF). Kazdy procesor ma na pocatku pripraven paket o velikosti m. Pak popiste co nejefektivnejsi algoritmus pro AAB, urcete jeho pocet kroku rAAB(Qn,m) a komunikacni zpozdeni TAAB(Qn,m). Predpokladejte, ze prenos paketu o velikosti ny mezi uzly ve vzdalenosti d trva t(d,ny)=ts+ny*d*tm. Casovou rezii na rozklad zprav a kombinovani paketu do zprav neuvazujte.

2. [3b] odvodte spodni meze na pocet kroku roAAS a na komunikacni zpozdeni tauAAS nekombinujiciho rozesilani AAS na 3D T(z1,z2,z3), z1>z2>z3 s plne duplexnimi kanaly a WH. Rychlost kanalu je tm, kazdy uzel ma pro kazdy jiny uzel paket o velikosti m

3. [3b] Spodni mez na pocet kroku AAS a komunikacni zpozdeni AAS nekombinujiciho rozesilani vsichni vsem AAS na 3D toroidu (z1, z2, z3) z1>z2>z3 s plne duplex kanaly, WH, jestlize rychlost kanalu je tm[B/s] a kazdy uzel ma pro kazdy jiny uzel paket o velikosti m[B].

4. AAS BEX (binarni vymena) M(odm(n),odm(n)) Urcit pocet kroku a cas
*Pocet kroku je log odm(N)*log odm(N)=log n   slozitost je nekde ve skriptech*

5. AAS nebo AAB - presne nevim, nestihl jsem ani precist

6. [4b] OAS na toroidu K(z1,z2), 1-port, WH, bod (a1,a2), 1 paket o vel. m, t = ts+d*td+mi*tm, efekt alg., Ro, tau = ?, r, t =? porovnat optimalitu

7. OAB na 2-port 1D Mrizce

8. [4b] Odvodte **spodni mez** na pocet kroku ρ_{AAB}(K(z,z)) a casovou slozitost τ_{AAB}(K(z,z),m) pro **vysilani vsichni vsem (AAB)** na **vse-portovem, plne-duplexnim nekombinujicim toroidu** K(z,z), kde z >= 5 je **liche** cislo a kazdy uzel ma na pocatku pripraven paket o velikosti m. Predpokladejte prepinani paketu uloz-posli-dal (SF), kde prenos paketu o velikosti μ mezi 2 uzly ve vzdalenosti d trva t(d,μ) = $t_s + \mu d t_m$. Popiste co nejlepsi algoritmus pro tento komunikacni problem a odvodte jeho **pocet kroku**. r_{AAB}(K(z,z)) a **casovou slozitost** t_{AAB}(K(z,z),m) a vyslovte tvrzeni o optimalite vaseho reseni. /4b/

---

a. [5b] Odvodte spodni meze pro pocet kroku a casovou slozitost u OAS 1-port WH 2D mrizka M(z1, z2), zdroj vysilani na [a1, a2] ma pro kazdy uzel pripravenu zpravu o velikosti M, predpokladejte ze prenos zpravy o velikosti m na vzdalenost d trva t(d, m) = ts + dtd +mtm
Popiste co nejefektivnejsi OAS algoritmus, urcete pocet kroku a casovou slozitost a porovnejte se spodnimi mezemi

b. [1b] Jaka je spodni mez poctu kroku AAB na Qn? Odvodte... (napsal sem, ze je to n, jako ze prumer grafu a to je blbe, vic nevim.)

c. [4b] Napiste optimalni algoritmus pro OAB na M(z1, z2), jestli ze je 2-portova, plneduplexni, WH prepinani. Urcete spodni meze poctu kroku (nebo mozna casu) a casovou slozitost, pokud mezi sousedy trva t=ts+td+m*tm.

d. [4]mrizka m(z1,z2) a provedte multicast na M, jakou ma spodni mez na pocet kroku, popiste slovne algoritmus, ktery to provadi v optimalnim poctu kroku a ukazte na zadanem priklade
tento priklad neni tezky a vsem doporucuji na nej kouknout, a poradne promyslet

e. [5b] OAB 2D T, WH, all-port napsat spodní mez OAB
tady měla být zaručeně zobecněná diagonála, ale hodně lidí psalo rúzné půlení a podobné nesmysly

f. QEX - kvadrantova vymena AAS na M(sqrt(p),sqrt(p)). Tusim WH prepinani. Urcete tau (mez min poctu kroku), rAAS, tAAS, a dalsi veci.
Algoritmus nasleduje ve skriptech za binarni vymenou (neplest !). Za ideu mi dal 1 bod, ale stetil se, ze tam nemam alespon velikost paketu. Cele to nemel snad nikdo.

11 Par alg. pro lin algebru

1. [5b] Popiste a vysvetlete Cannonuv algoritmus paralelniho nasobeni ctvercovych odm(N) x odm(N) matic C=AB na vseportovem 2-D toroidu K(odm(p),odm(p)) procesoru, kde p<N. Odvodte co nejpresneji vyraz pro paralelni cas T(N,p), jestlize (1) smerovace provadeji cervi (WH) prepinani paketu, kde prenos paketu mezi 2 uzly ve vzdalenosti d trva t(d,ny)=ts+d*td+ny*tm, (2) prvky matic maji velikost m a (3) sekvencni lokalni nasobeni dvou matic o velikosti r x r trva cas alfa*r^3, kde m a alfa jsou konst. Odvodte izoefektivni fce Psi1(p) a Psi2(N).

2. [3b]Popiste co nejrychlejsi algoritmus pro **transpozici matice** A typu n x n na p-procesorove hyperkrychli s **cervim** (WH) **prepinanim**, je-li matice A mapovana na hyperkrychli **blokove-sachovnicove**. Odvodte presny pocet komunikacnich **kroku** (rounds) $r(n^2,p)$ a **casovou slozitost** algoritmu $T(n^2,p)$, jestlize prenos paketu obsahujiciho *k* prvku matice na vzdalenost *d* trva cas $t(d,\mu) = t_s + dt_d + \mu t_m$ a transpozice matice r x r na jednom procesoru trva cas *k*? /3b/

3. [5b] Algo. nasobeni (odmN odmN) matice A blokove sachovnicove namapovane na krychli Qlogp 1<=p<=N vektorem x odmN x 1 mapovanym blokove na podkrychli Qlogodmp. Vysledny vektor y=Ax bude po skonceni ulozen stejne jako na pocatku vektor x. Predpokladejte ze p je odmocnina 4.
1 port smerovace WH
prenos paketu o velikosti u mezi p ve vzdalenosti d trva t(d,p)=ts+dtd+utm
prvky maj velikost m
nasobeni matice rxr s vektorem rx1 na 1 procesoru trva (a krat r na druhou) - a je konstanta
chce T, E, psi1, psi2

4. Algoritmus pro vypocet tridiagonalni soustavy rovnic na 1D mrizce (p<n) s WH prepinanim. Urcit T, Psi. (sudo-licha redukce)

5. Jakobiho algoritmus (klasika, proste vsechno k tomu)

6. Transpozice matice na 2-D Mrizce WH

7. [5b] algoritmus nasobeni sqrt(N) x sqrt(N) matice blokove sachovnicove mapovane na hyperkrychli Qlogp, 1<=p<=N vektorem x(sqrt(N) x 1) mapovanym blokove na 1 podkrychli Qlog(sqrt(p)). Vysledny vektor y = Ax bude po skonceni vypoctu ulozen stejne jako na pocatku x. Pro jednoduchost predpokladejte, ze p je mocnina 4.
1portove smerovani s WH, prenos paketu o velikosti mikro mezi procesory ve vzdalenosti d ma cas t(d,n)=ts + d*td + mikro*tm. Dale predpokladejte, ze prvky matice a vektoru maji velikost m a nasobeni matice r*r vektorem r*1 na 1 procesoru trva cas alfa*r^2, alfa je konst. Odvodte vyrazy pro paral. cas T(N,P) a efektivnost E(N,P) a izoefektivni fce fi1(p), fi2(N)

8. Foxuv algoritmus nasobeni matic

---

a. [5b] Popsat jednu iteraci Jacobiho metody. Vic jsem si, bohuzel, nepoznamenal, akorat ze to bylo na jednorozmerne mrizce M(p) :-(

b. [3] Co nejrychlejsi algoritmus pro transpozici matice A typu m x n na p-proc. Q s WH je-li matice A namapovana blokove sachovnicove. Presny pocet komunik. kroku (rounds) r(n^2,p) a cas. slozitost T(n^2,p), jestlize premisteni paketu obsahujici k prvku matici na vzdalenost d trva cas tau(d,k) = ts + dtd + ktm a transpozice matice r x r na 1 procesoru trva cas r^2.
[] Napiště co nejrychlejší alg. pro transpozici matice Nxn na p proc. Qn s WH, je-li A mapována blokově-šachovnicově. Odvoďte přesný počet kom. kroků r(n^2,p) a čas. složitost T(n^2,p) jestliže přenos paketu osbahujícího k prvků matice na vzdálenost d trvá tau(d,k)=Ts+d*Td+k*Tm a transpozice matice RxR na 1 proc. zabere čas R^2.

c. [4b] Popiste Canonuv algoritmus na ctverecovem toroidu s p procesory pro nasobeni matic, urcere jeho casovou slozitost a pak asi jeste funkce psi1 a psi2.

d. [5] FOX na Q_log(p), urcit fi1,fi2, pametovou slozitost
fi1=sqrt(p),fi2=n^2, slozitost nevim, tipl sem 3n/p (3x velikost submatice pro nasobeni, mozna to je jen n/p pro presun submatic pri komunikaci(?))

12 Teoreie par. slozitosti

1. [1b] Co je NC trida a 3 priklady

2. [1b] {3x} definujte tridu P-uplnych problemu a uvedte 2 priklady P-uplnych problemu