UNIVERSITETI POLITEKNIK I TIRANËS
FAKULTETI I TEKNOLOGJISË SË INFORMACIONIT
DEPARTAMENTI I ELEKTRONIKËS DHE TELEKOMUNIKACIONIT

Algoritmikë

Punë Laboratori nr. 1

Punoi : Enrik Doçi ; Klajdi Gjoka

Ing. Telekomunikacioni II – A

20 Prill 2016

Funksione shtesë

1. int menu( ) – Funksioni menu sa herë që thërritet pastron programin nga çdo tekst , printon listen e funksioneve që mund të thërrasë përdoruesi, dhe vlerën që do të skanojë e kthen (përkatësisht variablit int answer në main( )
2. trace* createTrace(int nbpts) – Funksioni createTrace merr si argument një numër int nbpts dhe pasi deklaron një pointer-to-trace rezervon në memorie vend për një trace të ri , gjithashtu edhe për vectorët float time dhe float value me nbpts elemente. Funksioni kthen pointerin e trace të ri që krijoi . Në këtë mënyrë programi bëhet më i lehtë për tu lexuar duke eliminuar nevojën për përsëritjen e rreshtave të kodit gjatë alokimit të memories në pjesë të ndryshme të programit .
3. int fileLength(char * filename) – Funksioni fileLength lexon skedarin emrin e të cilit e merr si argument dhe kthen numrin e matjeve që janë ruajtur në të . Ky funksion përdoret gjatë deklarimit të struct trace që do të mbajë të dhënat që janë ruajtur në skedarin cell.txt, në mënyrë që në memorie të rezervohet aq vend sa duhet për të ruajtur të dhënat (në këtë mënyrë programi punon pa problem për çdo madhësi të skedarit cell.txt) . Ky funksion gjithashtu llogarit edhe intervalin e kohës int file_tmax dhe float file_dt të matjeve në skedarin cell.txt , të dy variabla global, në mënyrë që gjatë gjetjes së gabimit mes matjes dhe simulimit, funksioni i simulimit simutrace të thërritet me të njëjtët parametra, pa patur nevojë për futjen e tyre manualisht nga përdoruesi.
4. float errorTraceLimited(int n, trace uneTrace1, trace uneTrace2) - Ky funksion është gati i njëjtë me funksionin errorTrace por merr një argument më shumë , i cili lejon që vetëm int n vlerave të para të simulimeve ti llogaritet gabimi me vlerat e matura (marre nga cell.txt)

Optimizime te tjera : free(simulim) përdoret për të liruar vendin në memorie që zë *simulim përpara se simulim (pointer-to trace) ti alokojmë vend të ri në memorie , në mënyrë që memoria të menaxhohet sa më mire .

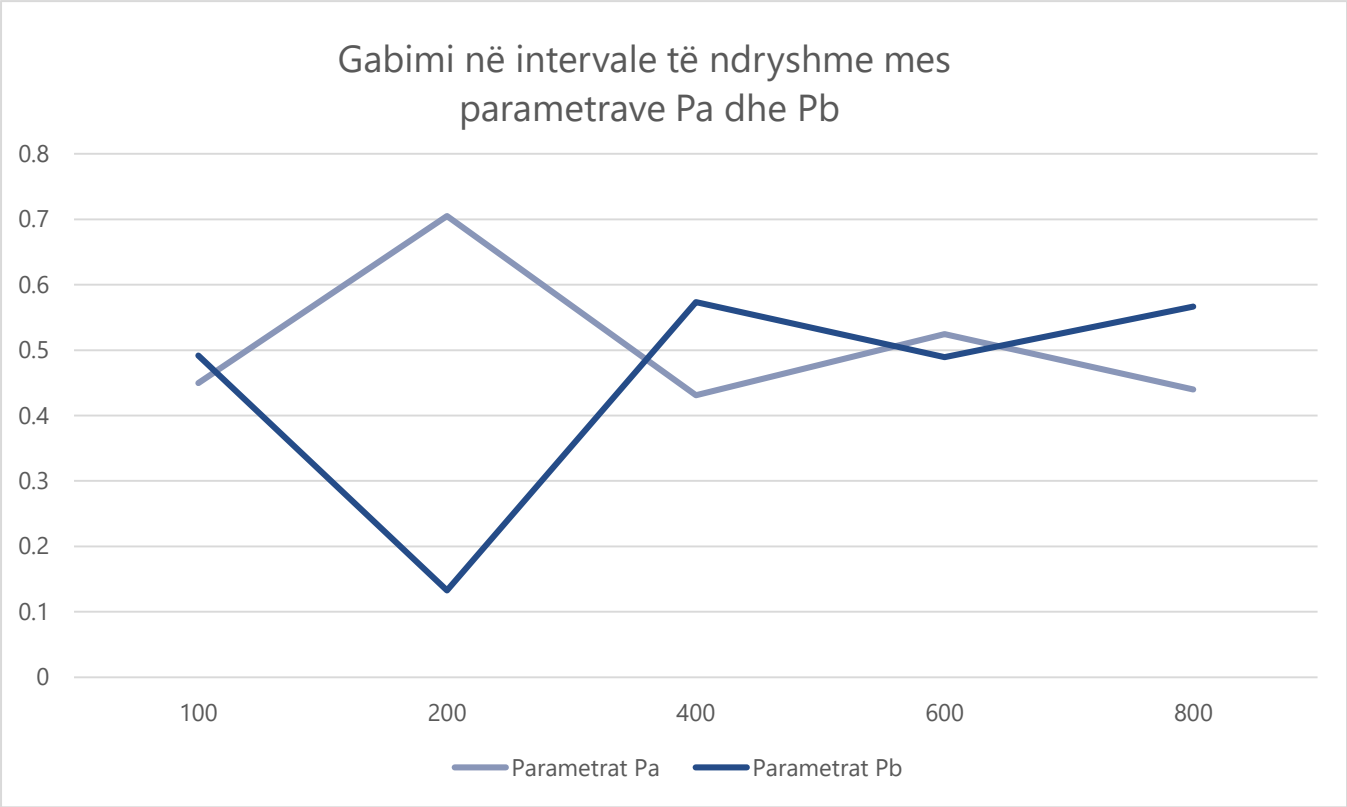
Probleme të hasura dhe zgjidhja e tyre

1. #IND00 – Gjatë gjetjes së gabimit vlera që printonte funksioni dilte #IND00 , vlerë e cila tregon vlerat e palejuara matematikore si pjesëtimi me zero ose rrënja katrore e një numri negativ. Zgjidhja në këtë rast ishte përdorimi i funksionit fabs(float) i cili sikurse abs(int) , kthen vlerën absolute të një numri me presje float.
2. Problemet e tjera të hasura si deklarimi i panevojshem ose i pamjaftueshëm i hapësirës në vectorët e përdorur , menaxhimi jo i mirë i memories , nevoja për të patur të njëjtin numër vlerash gjatë gjetjes së gabimit të matjeve laboratorike me ato analitike u zgjidhën duke krijuar funksionet e mësipërm dhe duke optimizuar ato aktualë në mënyrë që programi të jetë efikas, i lehtë për tu përdorur nga përdoruesi dhe të eliminohen rastet që mund të behej crash .

Llogaritja e gabimit

Pas thërritjes së funksionit errorTrace me vlerat e përftuara shohim që simulimi duke përdorur parametrat Pa është më i përafërt me vlerat e marra nga matjet eksperimentale se simulimi me parametrat Pb , me vlerën e gabimit dale përkatësisht Pa = 0.481187 dhe Pb = 0.530630 .

Duke kufizuar gabimin në vetem disa nga vlerat e para shohim që pikërisht në intervalet ku Pa ka gabim të madh , Pb e ka gabimin shumë të vogël , duke e bërë më të përshtatshëm në zona të veçanta , pra jo domosdoshmërisht parametrat Pa janë më të saktë në çdo eksperiment .



Për gjetjen e gabimit në interval prej 100 , 200, 400, 600 dhe 800 sekondash përdoret funksioni errorTraceLimited i cili ishte më i shpejti për tu implementuar dhe njekohësisht ka përsëri kompleksitet të rendit O(n), pra është gjithashtu efikas .

Llogaritja e kompleksitetit

Çdo funksion përveç main( ) , menu( ) dhe createTrace(int nbpts) janë funksione me vetëm një cikël të njëfishtë , pra kanë kompleksitet linear dhe bëjnë pjesë në klasën O , pra kompleksitetin e kane O(n)

Për shembull :

Simutrace përsëritrt tmax/dt herë , pra g(n) = tmax/dt*n (ku tmax dhe dt konstante) , dhe limiti kur n shkon në infinit i g(n)/f(n) (ku f(n) = n), jep konstanten tmax/dt , gjë që provon klasën e kompleksitetit dhe vlerën e tij siç u tregua më sipër .

Po njësoj vërtetohet klasa dhe kompleksiteti i funksioneve të tjerë linearë , të cilët janë të gjithë f(n)*c + konstante të tjera .

Funksionet createTrace dhe menu nuk kanë as cikle dhe as thërritje funksionesh të tjerë brenda tyre pra kompleksiteti i tyre është constant .

Përfundimisht kompleksiteti i gjithë programit është po prap O(n) si shumë e funksioneve të tjerë linearë .

Punoi : Klajdi Gjoka ; Enrik Doçi

Ing. Telekomunikacion II-A

20 Prill 2016