

Performanse računarskih Sistema

Domaći zadatak za junijski ispitni rok 2018/2019

Predmetni profesor: dr Jelica Protić

Opis sistema

Multiprogramski računar ima procesor, dva systemska diska i K korisničkih diskova. Procesorska obrada traje u proseku $S_p = 5ms$, prosečno vreme opsluživanja za systemske diskove iznosi po $S_{d1} = 12ms$, $S_{d2} = 15ms$, a za korisničke diskove po $S_{dk} = 20ms$. Sva vremena imaju eksponencijalnu raspodelu. Posle procesorske obrade u 15% slučajeva zahteva se pristup prvom sistemskom disku, u 15% slučajeva drugom, a u 70% slučajeva, pristupa se nekom od K korisničkih diskova (sa jednakom verovatnoćom za svaki disk). Posle pristupa nekom sistemskom disku, u 50% slučajeva ne vrši se nikakva obrada već se procesi ponovo vraćaju u procesorski red. U preostalih 50% slučajeva pristupa se nekom od K korisničkih diskova (sa jednakom verovatnoćom za svaki disk). Posle pristupa korisničkom disku proces se vraća u procesorski red. Procesori imaju jedinstven red za čekanje i podjednako su opterećeni.

Za zatvorenu mrežu kojom se modelira ovaj računarski sistem, potrebno je odrediti iskorišćenja resursa, protoke kroz resurse, prosečan broj poslova u svakom od resursa ovog sistema i vreme odziva sistema (prosečno trajanje jednog ciklusa, od kada proces dođe u procesorski red, dok ne dođe u taj red sledeći put) za Kod 2 do 8. Ovi parametri se određuju za stepen multiprogramiranja od 10, 15 i 25. Odrediti kritični resurs u sistemu.

Zadaci

- 1) Potrebno je napisati program koji će za ulazni parametar programa n (stepen multiprogramiranja):
 - a) (13p) Simulirati dati sistem i na osnovu rezultata simulacije odrediti i prikazati (upisati u prvi izlazni fajl) sve tražene parametre sistema za svako K (broj korisničkih diskova). Potrebno je obezbediti mogućnost podešavanja simuliranog vremena rada sistema (u minutima), a podrazumevano vreme trajanja simulacije je 18h (ne realnog vremena rada programa, već simuliranog vremena rada sistema!).
 - b) (5p) Obezbediti funkcionalnost koja na osnovu definisanih ulaznih parametara sistema Gordon-Njuelovom metodom određuje normalizovane vremenske potražnje servera (normalizaciju izvršiti u odnosu na procesor). Izračunavanje normalizovanih potražnji sprovesti za svako K (broj korisničkih diskova) i normalizovane potražnje upisati u drugi izlazni fajl.
 - c) (5p) Bjuzenovom metodom analitički odrediti sve tražene parametre i upisati ih u treći izlazni fajl. Analizu sprovesti za svako K (broj korisničkih diskova).
 - d) (2p) Odrediti relativno odstupanje (u procentima) rezultata simulacione metode od analitičke. Ova odstupanja je potrebno izračunati za svaki broj korisničkih diskova K_i upisati ih u zaseban (četvrti) izlazni fajl.
- 2) (5p) Potrebno je napisati dokumentaciju koja detaljno objašnjava metod simulacije i analitičko rešavanje problema. U dokumentaciji priložiti izveštaje relativnog odstupanja traženih parametara sistema koji prikazuju vrednosti ovih parametara u zavisnosti od broja korisničkih diskova u sistemu za sledeće vrednosti parametra n : 10, 15, 20 poslova.

Arhitektura sistema

Zadatak je urađen u programskom jeziku Python. Osnovu programa čine:

1. Inicijalizacija skupa različitih brojeva korisničkih diskova i skupa različitih brojeva poslova za simulaciju
2. Odabiranje dužine, pokretanje simulacije i prihvatanje svih izračunatih vrednosti, zatim snimanje istih u fajl
3. Pokretanje analitičkog izračunavanja sa istim parametrima kao i za simulaciju i snimanje rezultata u fajl u istom formatu
4. Upoređivanje rezultata simulacije i analitičkog izračunavanja i ispisivanje u fajl u istom formatu, kao i iscrtavanje grafikona zavisnosti svih praćenih vrednosti od broja korisničkih diskova i broja poslova

Simulacija

Zadaje se željena dužina trajanja simulacije. Za svaki broj korisničkih diskova i za svaki broj procesa u sistemu se pokreće simulacija. Vrednosti koje se prate kroz simulaciju su:

- Vreme koje je proteklo od početka simulacije (služi za uslov završetka simulacije)
- Broj koliko puta su svi procesi zajedno završili ceo prolaz kroz sistem
- Niz objekata resursa sistema koji predstavlja stanje sistema u nekom trenutku

Objekat svakog resursa se inicijalizuje na početku simulacije i ima sledeći sadržaj:

- Prosečno vreme obrađivanja procesa unutar tog resursa (zadato u postavci)
- Broj procesa koji čekaju u redu na tom resursu (jednak ukupnom broj poslova za procesor i nula za ostale resurse)
- Vremenski trenutak u kome će obrada procesa koji se trenutno obrađuje u resursu biti završena, ili -1 kao pokazatelja da je resurs slobodan
- Akumulator U koji broji koliko vremena je resurs proveo obrađujući neki proces
- Akumulator J koji broji koliko je u svakom vremenskom intervalu Δt procesa čekali i bilo obrađivano na tom resursu

Nakon inicijalizacije resursa jedan proces se uklanja iz reda za čekanje procesora i procesor počinje da ga izvršava. Određuje se vreme kraja tog izvršavanja nasumičnim dohvatanjem iz eksponencijalne distribucije koja ima srednju vrednost jednaku srednjem vremenu obrađivanja procesa u procesoru. Nakon toga ulazi se u petlju koja traje dokle god akumulirano vreme izvršavanje simulacije nije veće od zadate dužine simulacije. Svaka iteracija petlje se sastoji iz sledećih koraka:

- Pronalaženje resursa koji će najskorije završiti obradu procesa koji obrađuje, prolazenjem kroz niz resursa.
- Izračunava se vreme koje je preteklo od prethodne iteracije petlje (Δt) oduzimanjem vremena završetka prethodne obrade od vremena završetka trenutne obrade. Ažurira se trenutno vreme obrade.
- Prolaskom kroz niz resursa ažuriraju se akumulatori iskorišćenja (U) i prosečnog broja procesa na resursu (J). U se ažurira tako što se, ukoliko je resurs obrađivao neki proces tokom prethodnog perioda, na njega sabira Δt . J se ažurira tako što se pod istim uslovom na njega sabira broj resursa u njegovom redu za čekanje i još jedan za resurs koji je bio obrađivan, sve pomnoženo sa Δt , zato što duži vremenski intervali nose veću težinu u prosečnoj vrednosti.
- U zavisnosti od resursa koji je upravo završio obradu, probabilistički, po uputstvu iz postavke, se odabira kojem resursu će obrađen proces biti prosleđen

- Ukoliko je odabrani sledeći resurs procesor, dodaje se jedan na brojač završenih prolaska kroz sistem
- Proverava se da li resurs koji je upravo završio obradu i resurs kome je proces prosleđen imaju neprazan red za čekanje i da li nemaju već dugi proces u obradi. Ukoliko su oba uslova ispunjena, resurs počinje da obrađuje sledeći proces iz reda za čekanje i dodeljuje mu se vreme završetka obrade kao trenutno vreme plus nasumična vrednost iz eksponencijalne distribucije sa adekvatnom srednjom vrednošću.

Nakon što je dužina izvršavanja simulacije dostigla zadatu vrednost petlja se završava. Vršiti se ažuriranje statistike za resurse. Vrednosti za U i J se dele sa vremenom trajanja simulacije i dobijaju se ispravne srednje vrednosti iskorišćenja i broja procesa. Protok kroz resurs se računa kao U podeljeno sa prosečnim vremenom obrade procesa na tom resursu. Vreme odziva sistema se računa tako što se vreme simulacije pomnoži sa brojem procesa u sistemu i podeli sa brojem prolaza kroz sistem svih procesa zajedno. Svi izračunati podaci se vraćaju main funkciji. Odatle se podaci prosleđuju funkciji za formatiran ispis u fajl.

Analitičko rešenje

Nakon simulacije pokreće se analitičko izračunavanje Gordon-Newell-ovom metodom. Analitičko izračunavanje se sastoji iz sledećih koraka:

- Inicijalizacija prazne matrice dimenzija broj resursa puta broj resursa
- Eksplicitno ažuriranje nenultih vrednosti u matrici u skladu sa šemom sistema i verovatnoćama prelaza iz jednog resursa u druge
- Generisanje Gordon-Newell-ovih jednačina oduzimanjem jedinične matrice od prethodno dobijene matrice, zatim deljenjem svih kolona matrice prosečnim vremenom obrade procesa u resursu koji ta kolona matrice predstavlja.
- Pretpostavkom da se vrši normalizacija za $x_1 = 1$, prva kolona se anulira, a vrednosti koje su se u njoj nalazile i koje bi množile x_1 se prebacuju sa druge strane znaka jednakosti. Iz matrice se izbacuju anulirana kolona i prvi red (odnosno prva jednačina), zato što sada sistem ima jednu manje nepoznatu.
- Pomoću metode solve iz numpy linalg biblioteke, izračunava se rešenje sistema jednačina.
- Izračunava se niz Buzenovih brojeva pomoću rešenja sistema.
- Izračunavaju se iskorišćenosti za sve resurse množenjem rešenja sistema se količnikom pretposlednjeg i poslednjeg Buzenovog broja.
- Izračunavaju se protoci za sve resurse deljenjem iskorišćenosti sa njihovim prosečnim trajanjem obrade procesa.
- Izračunava se prosečan broj procesa u resursima trikom objašnjenim na vežbama:

$$\begin{aligned}
 n_p &= \sum_{m=1}^n m \cdot P[n_p=m] \\
 P[n_p \geq 1] &= P[n_p=1] + P[n_p=2] + P[n_p=3] + \dots + P[n_p=n] \\
 P[n_p \geq 2] &= P[n_p=2] + P[n_p=3] + \dots + P[n_p=n] \\
 P[n_p \geq 3] &= P[n_p=3] + \dots + P[n_p=n] \\
 &\dots \\
 P[n_p \geq n] &= P[n_p=n] \\
 \bar{n}_p &= P[n_p \geq 1] + P[n_p \geq 2] + P[n_p \geq 3] + \dots + P[n_p \geq n] = \\
 &= \sum_{m=1}^n x_p \cdot \frac{G(n-m)}{G(n)}
 \end{aligned}$$

- Izračunava se vreme odziva sistema kao broj poslova u sistemu podeljen sa zbirom vrednosti tokova koji utiču u red za čekanje procesora.

Svi podaci se vraćaju u main funkciju i prosleđuju u funkciju koja ih upisuje u fajl u istom formatu kao i rezultate simulacije, kao i funkciji koja u fajl upisuje rezultate rešenog linearnog sistema jednačina.

Rezultati simulacije i analitičkog izračunavanja se prosleđuju u metodu koja pronalazi relativno odstupanje tih rezultata u procentima i to još jednom ispisuje u fajl sa istom strukturom.

Poziva se i metoda koja rezultate simulacije i analitičkog izračunavanja iscrtava na 3D bar-
 grafovima na kojima su na x i y osama broj resursa i broj poslova u sistemu, visina na grafu označava
 vrednost Određenog izračunatog parametra određenog resursa.

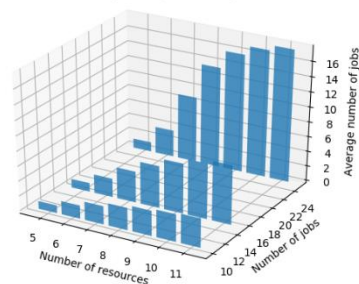
Relativna odstupanja u procentima:

	K = 2, N = 10	K = 2, N = 15	K = 2, N = 25	K = 3, N = 10	K = 3, N = 15	K = 3, N = 25
processor:						
utilization:	0.04749	-0.150761	0.032864	0.039626	0.126154	-0.008339
flow:	0.04749	-0.150761	0.032864	0.039626	0.126154	-0.008339
average jobs:	0.083313	-0.225482	0.099876	0.07646	0.253101	-0.001464
system disk 0:						
utilization:	0.054862	-0.423821	-0.310199	-0.269323	0.086382	0.07408
flow:	0.054862	-0.423821	-0.310199	-0.269323	0.086382	0.07408
average jobs:	0.036977	-0.408291	-0.316246	-0.338472	0.061186	0.08794
system disk 1:						
utilization:	-0.094753	-0.255396	-0.16192	0.049561	-0.042109	0.012828
flow:	-0.094753	-0.255396	-0.16192	0.049561	-0.042109	0.012828
average jobs:	-0.033933	-0.213096	-0.355898	-0.043432	-0.138839	-0.220727
user disk 0:						
utilization:	0.10012	-0.05755	0.054064	0.001219	0.14018	0.01416
flow:	0.10012	-0.05755	0.054064	0.001219	0.14018	0.01416
average jobs:	-0.283318	-0.387608	0.351874	-0.16898	0.163091	0.185019
user disk 1:						
utilization:	-0.078605	-0.040531	-0.041581	0.036042	-0.098984	-0.080983
flow:	-0.078605	-0.040531	-0.041581	0.036042	-0.098984	-0.080983
average jobs:	-1.005189	-0.062944	-0.514166	-0.023307	-0.707132	-0.046454
reaction time:	-0.001592	0.166392	0.028484	-0.005691	-0.088218	0.012169
	K = 4, N = 10	K = 4, N = 15	K = 4, N = 25	K = 5, N = 10	K = 5, N = 15	K = 5, N = 25
processor:						
utilization:	-0.011106	-0.030677	0.006614	-0.019929	0.004892	-0.009293
flow:	-0.011106	-0.030677	0.006614	-0.019929	0.004892	-0.009293
average jobs:	-0.076883	-0.038322	-0.045954	-0.034478	-0.112766	-0.189193
system disk 0:						
utilization:	0.006042	0.005826	-0.141009	-0.093539	0.031691	-0.0279
flow:	0.006042	0.005826	-0.141009	-0.093539	0.031691	-0.0279
average jobs:	-0.07834	0.013075	-0.169923	-0.038853	-0.030545	0.000982
system disk 1:						
utilization:	0.075939	-0.08078	0.001425	-0.097409	0.089698	-0.1845
flow:	0.075939	-0.08078	0.001425	-0.097409	0.089698	-0.1845
average jobs:	0.035478	-0.151695	-0.102003	-0.169211	0.01005	-0.53483
user disk 0:						
utilization:	-0.153391	0.042201	-0.062429	-0.103562	-0.080005	-0.028952
flow:	-0.153391	0.042201	-0.062429	-0.103562	-0.080005	-0.028952
average jobs:	-0.438173	0.124994	-0.018235	-0.139461	-0.160112	0.294893

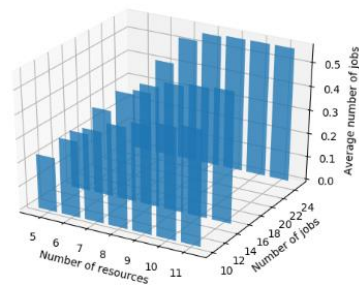
user disk 1:						
utilization:	0.13063	0.063971	-0.08728	-0.113665	0.031665	0.063032
flow:	0.13063	0.063971	-0.08728	-0.113665	0.031665	0.063032
average jobs:	0.183068	0.181512	-0.134205	-0.174311	0.173094	0.445812
reaction time:	-0.016173	0.012158	0.015003	0.006927	-0.054368	0.00562
	K = 6, N = 10	K = 6, N = 15	K = 6, N = 25	K = 7, N = 10	K = 7, N = 15	K = 7, N = 25
processor:						
utilization:	-0.054973	0.031557	0.00082	-0.014483	0.006674	0.0005
flow:	-0.054973	0.031557	0.00082	-0.014483	0.006674	0.0005
average jobs:	-0.208058	0.10308	0.021397	-0.149345	0.042571	-0.050835
system disk 0:						
utilization:	-0.079321	-0.115515	-0.201405	-0.11893	-0.198468	-0.001359
flow:	-0.079321	-0.115515	-0.201405	-0.11893	-0.198468	-0.001359
average jobs:	-0.102702	-0.196927	-0.17475	-0.191249	-0.179041	-0.106287
system disk 1:						
utilization:	-0.132178	-0.030399	0.107757	-0.22916	-0.028297	0.224113
flow:	-0.132178	-0.030399	0.107757	-0.22916	-0.028297	0.224113
average jobs:	-0.226035	0.066562	0.208273	-0.484698	0.087447	0.537605
user disk 0:						
utilization:	-0.018505	-0.029823	-0.114928	0.05295	-0.059699	0.123094
flow:	-0.018505	-0.029823	-0.114928	0.05295	-0.059699	0.123094
average jobs:	0.171987	-0.020276	-0.196309	0.006447	-0.134838	0.258893
user disk 1:						
utilization:	-0.05779	-0.044729	0.033546	-0.012789	-0.107551	-0.207054
flow:	-0.05779	-0.044729	0.033546	-0.012789	-0.107551	-0.207054
average jobs:	-0.122487	0.123052	0.137658	-0.132253	-0.114243	-0.131588
reaction time:	0.071349	-0.108234	0.137705	-0.099424	-0.155631	0.118792
	K = 8, N = 10	K = 8, N = 15	K = 8, N = 25			
processor:						
utilization:	-0.019914	0.003196	0.000314			
flow:	-0.019914	0.003196	0.000314			
average jobs:	-0.178518	-0.040859	-0.091378			
system disk 0:						
utilization:	0.043478	0.058502	0.171847			
flow:	0.043478	0.058502	0.171847			
average jobs:	0.088451	-0.031796	0.347135			
system disk 1:						
utilization:	0.162627	-0.05852	0.15494			
flow:	0.162627	-0.05852	0.15494			
average jobs:	0.16242	-0.028194	0.072585			
user disk 0:						
utilization:	0.062347	-0.002276	-0.030643			
flow:	0.062347	-0.002276	-0.030643			
average jobs:	0.072782	-0.114451	0.076181			
user disk 1:						
utilization:	-0.19287	-0.079623	0.005451			
flow:	-0.19287	-0.079623	0.005451			
average jobs:	-0.081725	-0.216502	0.204958			
reaction time:	-0.121756	0.12075	0.098256			

J

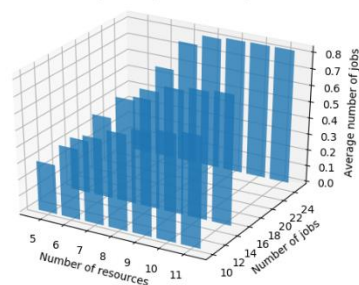
Analytical - processor - J



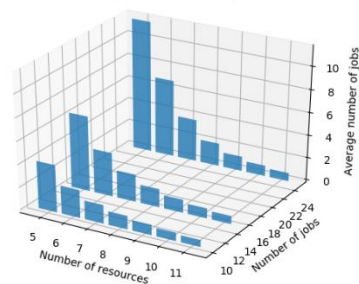
Analytical - system disk 0 - j



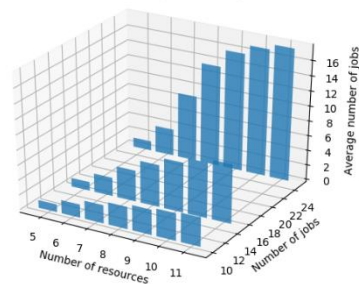
Analytical - system disk 1 - J



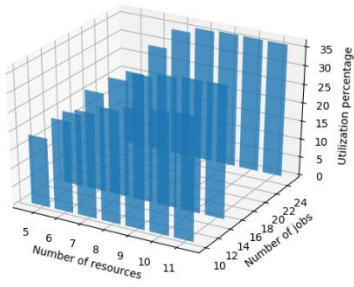
Analytical - user disks - J



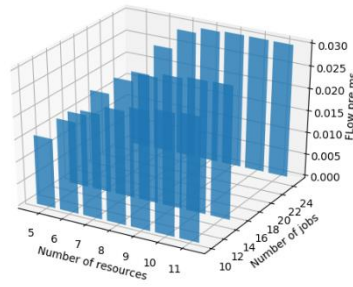
Simulation - processor - J



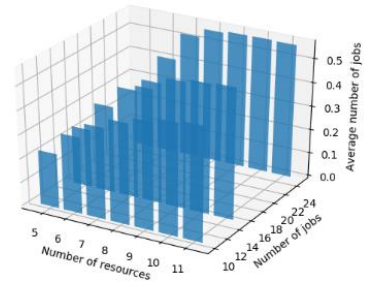
Simulation - system disk 0 - U



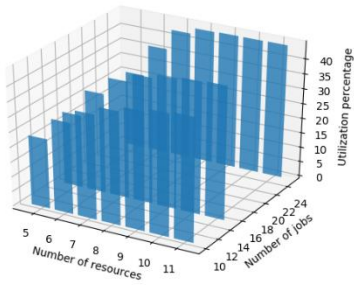
Simulation - system disk 0 - X



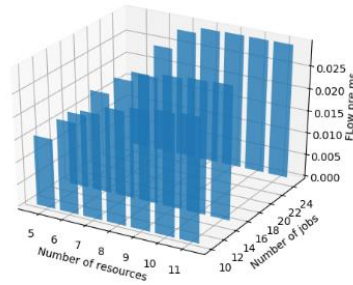
Simulation - system disk 0 - J



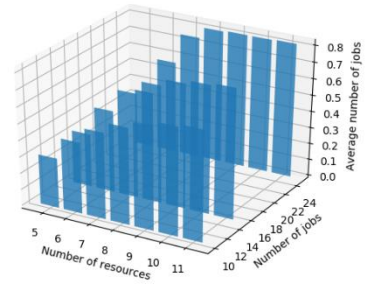
Simulation - system disk 1 - U



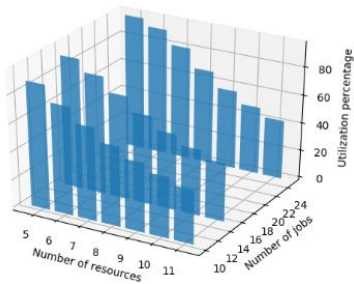
Simulation - system disk 1 - X



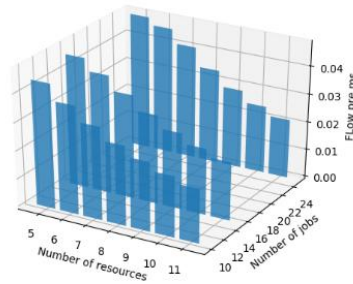
Simulation - system disk 1 - J



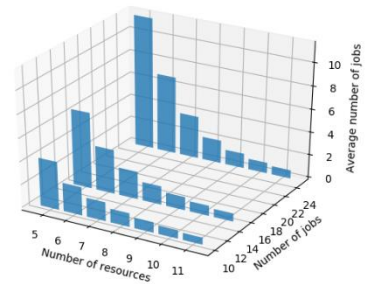
Simulation - user disks - U



Simulation - user disks - X

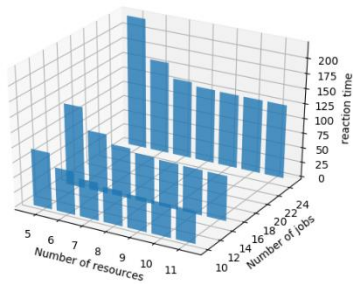


Simulation - user disks - J



R

Analytical - R



Simulation - R

