

GRILE SISTEME DE OPERARE

1.Care este numele comenzii care permite aflarea informatiilor despre TOATE attributele asociate unui fisier?

Raspuns:

2.Care este forma sintactica pentru executia unei comenzi avand fluxul stderr redirectionat catre un fisier, fara append?

- a. comanda parametri < fisier.txt
- b. comanda parametri > fisier.txt
- c.** comanda parametri 2 > fisier.txt
- d. comanda parametri 2 >> fisier.txt

3.Care este forma sintactica de compunere a doua comenzi simple pentru executia celei de a doua comenzi conditionata de esecul executiei primei comenzi?

- a. cmd1 ; cmd2
- b. cmd1 | cmd 2
- c. cmd1 & cmd2
- d.** cmd1 || cmd2
- e. cmd1 && cmd2

4.Care este apelul POSIX de I/O cu fisiere, utilizat pentru a crea un fisier de tip director?

- a. creat
- b. mkdir
- c.** mkdir
- d. mkfolder
- e. chroot

5.Se considera urmatoarea secventa de cod ce xecuta apelul POSIX stat pe fisierul program.c. Bifati doar optiunile ce sunt MINIM necesare astfel incat executia codului sa afiseze pe ecran textul "135".

```

struct stat f_metadata;
if(! stat("program.c",&f_metadata)) {
    if(f_metadata.st_mode & S_IWUSR) printf("1");
    if(f_metadata.st_mode & S_IRGRP) printf("2");
    if(f_metadata.st_mode & S_IXOTH) printf("3");
    if(f_metadata.st_mode & S_IWGRP) printf("4");
    if(f_metadata.st_mode & S_IRUSR) printf("5");
}

```

- a. proprietarul fisierului are drept de citire
- b. proprietarul fisierului are drept de scriere
- c. proprietarul fisierului nu are drept de executie
- d. colegii din grupul proprietar al fisierului nu au drept de citire
- e. colegii din grupul proprietar al fisierului nu au drept de scriere
- f. colegii din grupul proprietar nu au drept de executie
- g. utilizatorii care nu sunt in grupul proprietarului fisierului nu au drept de citire
- h. utilizatorii care nu sunt in grupul proprietarului fisierului nu au drept de scriere
- i. utilizatorii care nu sunt in grupul proprietarului fisierului nu au drept de executie

6.Care este apelul POSIX utilizat pentru aflarea ID-ului proprietarului efectiv al procesului apelant?

- a. getpid
- b. getppid
- c. getgid
- d. getuid
- e. geteuid
- f. getegid

7.Care este apelul POSIX ce permite crearea unui copil al procesului apelant?

- a. system
- b. wait
- c. fork
- d. oriare din functiile exec
- e. waitpid

8.Ce se intampla daca un proces executa un apel de scriere intr-un canal de comunicatie anonim, intr-un moment cand niciun proces nu are deschis capatul de citire din acel canal?

- a. apelul de scriere ramane blocat pana cand vreun alt proces va deschide capatul de citire al acelui canal
- ☒ b. sistemul notifica procesul despre aceasta situatie, printr-un semnal SIGPIPE
- c. apelul de scriere returneaza imediat si programul isi continua executia
- d. sistemul notifica procesul despre aceasta situatie, printr-un semnal SIGFIFO
- e. procesul este terminat fortat si se afiseaza mesajul "Segmentation fault"

9.Bifati toate apelurile POSIX ce pot fi utilizate pentru a asocia unui semnal o rutin proprie de tratare a sa.

- a. kill
- ☒ b. signal
- ☒ c. sigaction
- d. sighandler
- e. killall

10.Bifati TOATE secventele de cod prin care se redirecteaza iesirea normala standard catre capatul de scriere al unui canal anonim referit prin variabila p.

- ☒ a. close(1); dup2(p[1],1);
- ☒ b. close(1); dup(p[1]);
- c. close(0); dup2(p[0],0);
- d. close(0); dup(p[0]);
- e. close(1); dup(p[0],1);
- f. close(0); dup2(p[1],0);

11.Care este tipul de semnal ce este generat cand utilizatorul apasa combinatia de taste CTRL+C?

- a. SIGCONT
- b. SIGTSTP
- c. SIGSTOP
- ☒ d. SIGINT
- e. SIGQUIT

12.Se considera programul din figura alaturata. Bifati TOATE afirmatiile INCORECTE de mai jos. Observatii: 1.Se considera include toate declaratiile #include necesare. 2.Se presupune ca sectiunile de cod A,B si C nu contin apeluri exec sau exit(si nici return).

```

int main()
{
    /* Secventa de cod A */
    pid_t pid1;
    if(-1 == (pid1=fork())) )
    {
        perror("Eroare la fork"); return 1;
    }
    if(pid1 != 0)
    {
        /* Secvență de cod B */
    }
    else
    {
        /* Secvență de cod C */
    }
    /* Secvență de cod D */
    return 0;
}

```

- a. procesul tata va executa doar secventele de cod A si D
- b.** procesul fiu nu va executa secventa de cod B
- c. procesul fiu va executa doar secventele de cod B si C
- d.** procesul tata va executa doar secventa de cod A
- e. procesul tata va executa secventa de cod B
- f. procesul fiu nu va executa secventa de cod D

13. Clasificati urmatoarele sisteme de operare dupa numarul de utilizatori ce pot lucra simultan pe sistem.

| | mono-tasking | mono-utilizator | multi-tasking | multi-utilizator |
|----------------------------|--------------|-----------------|---------------|------------------|
| DOS | | | | |
| Windows NT server edition | | | | |
| Windows NT desktop edition | | | | |
| Linux | | | | |

14. Bifați TOATE obiectivele de planificare ce au caracter calitativ, nu cantitativ.

- ☒ a. echitate
- b. timpul de raspuns
- c. gradul de utilizare a CPU
- d. rata de servire
- ☒ e. evitarea infometarii
- f. indeplinirea termenelor limita

15. Bifați TOTI algoritmi de planificare care permit trecerea unui proces de la starea running direct la starea ready.

- a. FCFS
- b. SJF
- ☒ c. SRTF
- ☒ d. Algoritmul cu prioritati preemptiv
- e. Algoritmul cu prioritati nepreemptiv
- ☒ f. RR

16. Care dintre algoritmi urmasori se utilizeaza pentru a rezolva problema sectiunii critice, in cazul particular $n=2$?

- ☒ a. Algoritmul lui Peterson
- b. Algoritmul lui Dijkstra
- c. Algoritmul bancherului
- d. Algoritmul lui Courtios, Heymans si Parnas
- e. Niciunul dintre cele amintite

17. In ce context se utilizeaza algoritmul bancherului (i.e., ce problema rezolva acesta)?

- a. excluderea mutuala (i.e., problema sectiunii critice)
- b. acordarea unui imprumut bancar
- c. evitarea interblocajelor
- ☒ d. prevenirea interblocajelor
- e. detectia interblocajelor

18. Bifați TOATE strategiile ce se pot utiliza pentru rezolvarea fenomenului de infometare a proceselor.

- a. strategia de ignorare
- b. strategia de evitare
- ☒ c. strategia de prevenire
- ☒ d. strategia de detectie si recovery

- e. strategia FIFO
- f. strategia bazata pe aging

19.Cum se numeste algoritmul folosit pentru alocarea dinamica a memoriei, care selecteaza spatiul liber de memorie, de dimensiune maxima, ce este suficient de mare in raport cu cererea primita?

- a. FFA
- b. BFA
- c. WFA
- d. Alocatorul buddy-system
- e. Niciunul dintre cele amintite

20.Bifati TOATE tehnicile de administrare a memoriei principale care sufera de fenomenul de fragmentare externa.

- a. Alocarea contigua a memoriei in partitii fixe
- b. Alocarea contigua a memoriei in partitii variabile
- c. Segmentare
- d. Paginare
- e. Segmentare paginata

21.Cum se numeste algoritmul folosit pentru page-swapping, care selecteaza drept vitcima o pagina care a fost cel mai putin accesata in trecut (de la inceperea executiei programului respectiv)?

- a. NRU
- b. FIFO
- c. LRU
- d. LFU
- e. MFU

22.Care este formula timpului efectiv de acces la memorie pentru o arhitectura hardware precum x86(Intel/AMD pe 32 bits), ce foloseste tabele de paginare ierarhice cu 2 nivele?

- a. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio})(\text{TLB_AT} + 1 * \text{Memory_AT}))$
- b. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio})(\text{TLB_AT} + 2 * \text{Memory_AT}))$
- c. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio})(\text{TLB_AT} + 3 * \text{Memory_AT}))$
- d. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio})(\text{TLB_AT} + 4 * \text{Memory_AT}))$
- e. $2 * \text{Memory_AT}$
- f. $3 * \text{Memory_AT}$
- g. $4 * \text{Memory_AT}$

23.Care dintre urmatoorii algoritmi este folosit pentru optimizarea latentei de rotatie a acceselor la disc?

- a. FCFS
- b. SSTF**
- c. SRTF
- d. SLTF
- e. SCAN
- f. LOOK

24.Care dintre urmatoorii algoritmi NU este folosit pentru optimizarea acceselor la disc?

- a. FCFS
- b. SSTF
- c. SRTF**
- d. SLTF
- e. SCAN
- f. LOOK

25.Clasificati urmatoarele sisteme de fisiere dupa modul de utilizare.

| | de uz general | de uz particularizat |
|--------|---------------|----------------------|
| NTFS | | |
| tmpfs | | |
| ext4fs | | |
| procfs | | |
| brtfs | | |

26.Se considera un sistem de calcul SMP cu trei procesoare, ce utilizeaza pentru planificare algoritmul RR avand configurata cuanta = 3 unitati de timp, cu o singura coada ready (i.e., folosita in comun pentru cele trei procesoare). Ordinea de alegere a procesoarelor: la RR, daca la un moment dat mai multe procesoare sunt in idle(i.e., libere), pentru a planifica un proces ready planificatorul va alege procesorul cu cel mai mic ID dintre acestea; iar in caz contrar, daca toate procesoarele sunt ocupate si macar unul este preemptibil, pentru a planifica un proces ready planificatorul va alege procesorul cu cel mai mic ID dintre cele care sunt preemptibile. Cinci joburi A-E (ce nu efectueaza operatii I/O) asteapta sa ruleze in sistem. Timpii lor de rulare si momentele de intrare in sistem sunt specificate in

tabelul de mai jos. Planificati executia celor cinci joburi in cadrul sistemului dat, raspunzand la intrebarile de mai jos.

| Job | Sosire | Serviciu |
|-----|--------|----------|
| A | 1 | 6 |
| B | 0 | 7 |
| C | 2.001 | 4 |
| D | 2 | 6 |
| E | 5.001 | 2 |

26.1.Pentru fiecare proces, bifati TOATE intervalele de timp in care acel proces ruleaza pe primul procesor CPU1.

[illegible]

26.2.Pentru fiecare proces, bifati TOATE intervalele de timp in care acel proces ruleaza pe al doilea procesor CPU2.

[illegible]

| | | | | | | | | | |
|---|--|--|--|--|--|--|--|--|--|
| E | | | | | | | | | |
|---|--|--|--|--|--|--|--|--|--|

26.3. Pentru fiecare proces, bifati TOATE intervalele de timp in care acel proces ruleaza pe al treilea procesor CPU3.

| | [0,1] | [1,2] | [2,3] | [3,4] | [5,6] | [6,7] | [7,8] | [8,9] | [9,10] |
|---|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| A | | | | | | | | | |
| B | | | | | | | | | |
| C | | | | | | | | | |
| D | | | | | | | | | |
| E | | | | | | | | | |

26.4. Pentru procesul B, bifati TOATE intervalele de timp in care este pastrat in coada ready.

- a. [0,1]
- b. [1,2]
- c. [2,3]
- d. [3,4]
- e. [4,5]
- f. [5,6]
- g. [6,7]
- h. [7,8]
- i. [8,9]
- j. [9,10]

26.5. Pentru procesul C, bifati TOATE intervalele de timp in care este pastrat in coada ready.

- k. [0,1]
- l. [1,2]
- m. [2,3]
- n. [3,4]
- o. [4,5]
- p. [5,6]
- q. [6,7]
- r. [7,8]
- s. [8,9]

t. [9,10]

26.6.Ce proces are rata de penalitate cea mai mare? Dar cea mai mica?

| | A | B | C | D | E |
|---------------------------------|---|---|---|---|---|
| Rata de penalitate cea mai mare | | | | | |
| Rata de penalitate cea mai mica | | | | | |

27.Care este numele comenzii care permite aflarea informațiilor despre utilizarea diverselor comenzi interne?

Raspuns:.....

28.Care este forma sintactică pentru execuția unei comenzi având fluxul stdout redirectionat către un fișier, cu append?

- a. comanda parametri > fisier.txt
- b. comanda parametri >> fisier.txt
- c. comanda parametri 2> fisier.txt
- d. comanda parametri 2>> fisier.txt

29.Care este forma sintactică de compunere a două comenzi simple pentru execuția celei de a doua comenzi condiționată de succesul execuției primei comenzi?

- a. cmd1 ; cmd2
- b. cmd1 | cmd2
- c. cmd1 & cmd2
- d. cmd1 || cmd2
- e. cmd1 && cmd2

30.Care este apelul POSIX de I/O cu fișiere, pe care îl utilizăm pentru a citi informații dintr-un fișier de tip obișnuit?

- a. fscanf
- b. fprintf
- c. read
- d. scanf
- e. fread

31. Se consideră următoarea secvență de cod ce execută apelul POSIX `stat` pe fișierul `program.c`. Bifați doar opțiunile ce sunt MINIM necesare astfel încât execuția codului să afișeze pe ecran textul "234".

```
struct stat f_metadata;
if(! stat("program.c",&f_metadata)) {
    if(f_metadata.st_mode & S_IWUSR) printf("1");
    if(f_metadata.st_mode & S_IRGRP) printf("2");
    if(f_metadata.st_mode & S_IXOTH) printf("3");
    if(f_metadata.st_mode & S_IWGRP) printf("4");
    if(f_metadata.st_mode & S_IRUSR) printf("5");
}
```

- a. Proprietarul fișierului are drept de citire.
- b. Proprietarul fișierului are drept de scriere.
- c. Proprietarul fișierului nu are drept de execuție.
- d. Colegii din grupul proprietar al fișierului nu au drept de citire.
- e. Colegii din grupul proprietar al fișierului nu au drept de scriere.
- f. Colegii din grupul proprietar al fișierului au drept de execuție.
- g. Utilizatorii care nu sunt în grupul proprietarului fișierului nu au drept de citire.
- h. Utilizatorii care nu sunt în grupul proprietarului fișierului au drept de scriere.
- i. Utilizatorii care nu sunt în grupul proprietarului fișierului au drept de execuție

32. Care este apelul POSIX care suspendă execuția unui proces până când unul dintre copii acestuia, specificat prin PID-ul său, își termină execuția?

- a. `getpid`
- b. `sleep`
- c. `waitpid`
- d. `wait`
- e. `suspend`

33. Care este funcția ce permite lansarea de comenzi UNIX dintr-un program C, în maniera următoare: se creează un nou proces în care se încarcă interpretorul implicit, iar acesta va executa comanda specificată?

- a. `system`
- b. `wait`
- c. `fork`
- d. `execv`
- e. `runcmd`

34.Ce se întâmplă dacă un proces execută un apel de citire dintr-un canal de comunicație fifo, într-un moment când niciun proces nu are deschis capătul de scriere în acel canal?

- a. apelul de citire rămâne blocat până când vreun alt proces va deschide capătul de scriere al acelui canal
- b. sistemul notifică procesul despre această situație, printr-un semnal SIGPIPE
- c. apelul de citire returnează imediat valoarea pentru EOF și programul își continuă execuția
- d. sistemul notifică procesul despre această situație, printr-un semnal SIGFIFO
- e. procesul este terminat forțat și se afișează mesajul "Segmentation fault"

35.BIFAȚI toate apelurile POSIX ce pot fi utilizate pentru a aștepta primirea unui semnal de către procesul apelant.

- a. sigsending
- b. sigpending
- c. sigsuspend
- d. sigprocmask
- e. pause

36.Bifați TOATE secvențele de cod prin care se redirectează intrarea standard către capătul de citire al unui canal anonim referit prin variabila p.

- a. close(1); dup2(p[1],1);
- b. close(1); dup(p[1]);
- c. close(0); dup2(p[0],0);
- d. close(0); dup(p[0]);
- e. close(1); dup(p[0],1);
- f. close(0); dup2(p[1],0);

37.Care este tipul de semnal ce este generat când utilizatorul apasă combinația de taste CTRL+\ ?

- a. SIGCONT
- b. SIGTSTP
- c. SIGSTOP
- d. SIGINT
- e. SIGQUIT

38.Se execută următorul program, în care presupunem că apelul fork() nu eșuează, iar programul „cmd” returnează valoarea 0 numai dacă se execută cu succes. Bifați

TOATE afirmațiile adevărate de mai jos. Observație: se consideră că programul are incluse toate declarațiile #include necesare.

```
int main(int argc, char *argv[])
{
    int status;
    switch( fork() )
    {
        case -1: perror("Eroare la fork"); return 1;
        case 0: execlp("cmd","cmd",...,NULL); printf("1"); break;
        default : wait(&status);
                if( WIFEXITED(status) ) {
                    switch( WEXITSTATUS(status) ) {
                        case 0 : printf("2"); break;
                        case 1 : printf("3"); break;
                        default: printf("4");
                    }
                }
                else printf("5");
                return 0;
    }
    return 2;
}
```

- Dacă apelul funcției execlp eșuează, atunci pe ecran se va afișa (și) cifra 1.
- Dacă apelul funcției execlp eșuează, atunci pe ecran se va afișa (și) cifra 4.
- Tatăl afișează (și) cifra 2, indiferent de succesul sau eșecul execuției apelului execlp.
- În cazul în care programul "cmd" și-a terminat execuția cu succes în procesul fiu, atunci tatăl afișează (și) cifra 3.
- În cazul în care programul "cmd" și-a terminat execuția cu insucces în procesul fiu, atunci tatăl afișează 14.
- În cazul în care programul "cmd" a fost terminat forțat în procesul fiu, atunci pe ecran se va afișa (și) cifra 5.

39. Clasificați următoarele sisteme de operare după numărul de programe executate simultan.

| | mono-tasking | mono-utilizator | multi-tasking | multi-utilizator |
|-----|--------------|-----------------|---------------|------------------|
| DOS | | | | |

| | | | | |
|-----------------|--|--|--|--|
| Windows NT | | | | |
| Windows 3.x/9.x | | | | |
| Linux | | | | |

40. Bifați TOATE obiectivele de planificare ce au un caracter cantitativ, nu calitativ.

- a. echitate
- b. timpul de răspuns
- c. gradul de utilizare a CPU
- d. rata de servire
- e. evitarea înfometării
- f. îndeplinirea termenelor limită

41. Bifați TOȚI algoritmi de planificare care NU permit trecerea unui proces de la starea running direct la starea ready.

- a. FCFS
- b. SJF
- c. SRTF
- d. Algoritmul cu priorități preemptiv
- e. Algoritmul cu priorități nepreemptiv
- f. RR

42. În ce context se utilizează algoritmul brutarului (i.e., ce problemă rezolvă acest algoritm)?

- a. excluderea mutuală (i.e., problema secțiunii critice)
- b. activitatea unei brutării
- c. evitarea interblocajelor
- d. prevenirea interblocajelor
- e. detecția interblocajelor
- f. rezolvarea problemei mulțimilor înfometate

43. Care dintre problemele de sincronizare studiate modelează șablonul de cooperare CREW?

- a. problema secțiunii critice
- b. problema producători-consumatori
- c. problema cititori și scriitori
- d. problema cinei filozofilor
- e. problema bărbierului adormit

f. Niciuna dintre cele amintite

44. Bifați TOATE strategiile ce se pot utiliza pentru rezolvarea fenomenului de interblocaj a proceselor.

- a. strategia de ignorare
- b. strategia de evitare
- c. strategia de prevenire
- d. strategia de detecție și recovery
- e. strategia FIFO
- f. strategia bazată pe aging

45. Cum se numește algoritmul folosit pentru alocarea dinamică a memoriei, care selectează spațiul liber de memorie, de dimensiune minimă, ce este suficient de mare în raport cu cererea primită?

- a. FFA
- b. BFA
- c. WFA
- d. Alocatorul buddy-system
- e. Niciunul dintre cele amintite

46. Bifați TOATE tehnicile de administrare a memoriei principale care suferă de fenomenul de fragmentare internă.

- a. Alocarea contiguă a memoriei în partiții fixe
- b. Alocarea contiguă a memoriei în partiții variabile
- c. Segmentarea
- d. Paginarea
- e. Segmentarea paginată

47. Cum se numește algoritmul folosit pentru page-swapping, care selectează drept victimă o pagină care nu a fost accesată în trecutul recent?

- a. NRU
- b. FIFO
- c. LRU
- d. LFU
- e. MFU

48. Care este formula timpului efectiv de acces la memorie pentru o arhitectură hardware ce folosește tabele de paginare ierarhice cu 3 nivele?

- a. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio}) * (\text{TLB_AT} + 1 * \text{Memory_AT}))$

- b. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio}) * (\text{TLB_AT} + 2 * \text{Memory_AT}))$
- c. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio}) * (\text{TLB_AT} + 3 * \text{Memory_AT}))$
- d. $1 * \text{Memory_AT} + (\text{Hit_ratio} * \text{TLB_AT} + (1 - \text{Hit_ratio}) * (\text{TLB_AT} + 4 * \text{Memory_AT}))$
- e. $2 * \text{Memory_AT}$
- f. $3 * \text{Memory_AT}$
- g. $4 * \text{Memory_AT}$

49. Care dintre nivelele RAID asigură o performanță mai bună a accesului la date, dar NU și o siguranță mai bună de păstrare a datelor?

- a. RAID 0
- b. RAID 1
- c. RAID 2
- d. RAID 3
- e. RAID 4
- f. RAID 5
- g. RAID 6

50. Care dintre următorii algoritmi folosiți pentru optimizarea acceselor la disc poate suferi fenomenul de înfometare?

- a. FCFS
- b. SSTF
- c. SCAN
- d. C-SCAN
- e. LOOK
- f. C-LOOK

51. Care este unitatea de alocare pe disc a unui fișier, pentru sistemele de fișiere de uz general?

- a. Bitul
- b. Octetul (8 biți)
- c. Sectorul (blocul-disc)
- d. Clusterul
- e. Niciuna dintre cele amintite

52. Ce metodă de alocare utilizează sistemul de fișiere ext4fs din Linux?

- a. alocare contiguă
- b. alocare înlănțuită
- c. alocare indexată
- d. niciuna dintre metodele amintite

53. Se consideră un sistem de calcul SMP cu trei procesoare, ce utilizează pentru planificare algoritmul RR având configurată cuanta = 2 unități de timp, cu o singură coadă ready (i.e., folosită în comun pentru cele trei procesoare). Ordinea de alegere a procesoarelor: la RR, dacă la un moment dat mai multe procesoare sunt idle (i.e., libere), pentru a planifica un proces ready planificatorul va alege procesorul cu cel mai mic ID dintre acestea; iar în caz contrar, dacă toate procesoarele sunt ocupate și măcar unul este preemptibil, pentru a planifica un proces ready planificatorul va alege procesorul cu cel mai mic ID dintre cele care sunt preemptibile. Cinci joburi A–E (ce nu efectuează operații I/O) așteaptă să ruleze în sistem. Timpii lor de rulare și momentele de intrare în sistem sunt specificate în tabelul de mai jos. Planificați execuția celor cinci joburi în cadrul sistemului dat, răspunzând la întrebările de mai jos.

| Job | Sosire | Serviciu |
|-----|--------|----------|
| A | 0 | 6 |
| B | 2.001 | 4 |
| C | 1 | 6 |
| D | 3.001 | 4 |
| E | 2.999 | 4 |

53.1. Pentru fiecare proces, bifați TOATE intervalele de timp în care acel proces rulează pe primul procesor CPU1.

| | [0,1] | [1,2] | [2,3] | [3,4] | [4,5] | [5,6] | [6,7] | [7,8] | [8,9] | [9,10] |
|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |

53.2. Pentru fiecare proces, bifați TOATE intervalele de timp în care acel proces rulează pe al doilea procesor CPU2.

| | [0,1] | [1,2] | [2,3] | [3,4] | [4,5] | [5,6] | [6,7] | [7,8] | [8,9] | [9,10] |
|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |

53.3. Pentru fiecare proces, bifați TOATE intervalele de timp în care acel proces rulează pe al treilea procesor CPU3.

| | [0,1] | [1,2] | [2,3] | [3,4] | [4,5] | [5,6] | [6,7] | [7,8] | [8,9] | [9,10] |
|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |

53.4. Pentru procesul C, bifați TOATE intervalele de timp în care este păstrat în coada ready.

- a. [0,1]
- b. [1,2]
- c. [2,3]
- d. [3,4]
- e. [4,5]
- f. [5,6]
- g. [6,7]
- h. [7,8]
- i. [8,9]
- j. [9,10]

53.5. Pentru procesul D, bifați TOATE intervalele de timp în care este păstrat în coada ready.

- k. [0,1]
- l. [1,2]
- m. [2,3]
- n. [3,4]
- o. [4,5]
- p. [5,6]
- q. [6,7]
- r. [7,8]
- s. [8,9]
- t. [9,10]

53.6.Ce proces are rata de penalitate cea mai mare, respectiv cea mai mică?

| | A | B | C | D | E |
|---------------------------------|---|---|---|---|---|
| Rata de penalitate cea mai mare | | | | | |
| Rata de penalitate cea mai mica | | | | | |

54.Bifati TOATE nivelele RAID care NU folosesc tehnica oglindirii pentru asigurarea unei sigurante mai bune de pastrare a datelor.

- a. RAID 1
- b. RAID (1 + 0)
- c. RAID 0
- d. RAID (0 + 1)
- e. RAID 5
- f. RAID 6

55.Care dintre algoritmii urmatoari se utilizeaza pentru a rezolva problema sectiunii critice, in cazul particular $n=2$?

- a. Algoritmul bancherului
- b. Algoritmul lui Courtois, Heymans si Parnas
- c. Algoritmul lui Dijkstra
- d. Niciunul dintre cele amintite
- e. Algoritmul lui Peterson

56.Care dintre urmtorii algoritmi folositi pentru optimizarea acceselor la disc poate suferi fenomenul de infometare?

- a. LOOK
- b. C-LOOK
- c. SSTF
- d. C-SCAN
- e. SCAN
- f. FCFS

57. Care este planificatorul responsabil cu alocarea de timp procesor proceselor ready?

- a. Niciunul dintre cele amintite
- b. Planificatorul pe termen scurt
- c. Planificatorul pe termen lung
- d. Planificatorul pe termen mediu

58. În ce context se utilizează algoritmul brutarului (i.e., ce problemă rezolvă acest algoritm)?

- a. prevenirea interblocajelor
- b. detectia interblocajelor
- c. activitatea unei brutarii
- d. excluderea mutuală (i.e., problema secțiunii critice)
- e. evitarea interblocajelor
- f. rezolvarea problemei multimedierii infometate

59. Care dintre următorii algoritmi NU este folosit pentru optimizarea acceselor la disc?

- a. LOOK
- b. SRTF
- c. SLTF
- d. SSTF
- e. SCAN
- f. FCFS

60. Ce metodă de alocare utilizează sistemul de fișiere btrfs din Linux?

- a. Alocarea contigua
- b. Alocarea în lanțuită
- c. Niciuna dintre cele amintite
- d. Alocarea indexată

61. Bifați TOATE strategiile ce se pot utiliza pentru rezolvarea fenomenului de infometare a proceselor.

- a. strategia FIFO
- b. strategia de detectie si recovery
- c. strategia de ignorare
- d. strategia de prevenire
- e. strategia bazata pe aging
- f. strategia de evitare

62. Clasificati urmatoarele sisteme de operare dupa numarul de utilizatori ce pot lucra simultan pe sistem.

| | mono-tasking | mono-utilizator | multi-tasking | multi-utilizator |
|----------------------------|--------------|-----------------|---------------|------------------|
| Windows NT server edition | | | | |
| Linux | | | | |
| DOS | | | | |
| Windows NT desktop edition | | | | |

63. Cum se numeste algoritmul folosit pentru alocarea dinamica a memoriei, care selecteaza un spatiu liber de memorie, de dimensiune putere a lui 2, ce este suficient de mare in raport cu cererea primita?

- a. WFA
- b. Alocatorul buddy-system
- c. BFA
- d. Niciuna dintre cele amintite
- e. FFA

64. Bifati TOTI algoritmi de planificare care NU permit trecerea unui proces de la starea running direct la starea ready.

- a. Algoritmul cu prioritati preemptiv
- b. RR
- c. SJF
- d. SRTF
- e. Algoritmul cu prioritati nepreemptiv
- f. FCFS

65. Bifați TOATE obiectivele de planificare ce au un caracter cantitativ, nu calitativ.

- a. îndeplinirea termenelor limita
- b. timpul de raspuns
- c. gradul de utilizare a CPU
- d. echitate
- e. evitarea infometarii
- f. rata de servire

66. Care este unitatea de alocare pe disc a unui fisier, pentru sistemele de fisiere de uz general?

- a. clusterul
- b. bitul
- c. niciuna dintre cele amintite
- d. octetul(8 biti)
- e. sectorul(blocul-disc)

67. Bifați TOATE tehnicile de administrare a memoriei principale care suferă de fenomenul de fragmentare externă.

- a. segmentarea
- b. alocarea contigua a memoriei în partitii fixe
- c. segmentarea paginată
- d. alocarea contigua a memoriei în partitii variabile
- e. paginarea

68. Cum se numește algoritmul folosit pentru page-sapping, care selectează drept victimă o pagină care a fost cel mai puțin accesată în trecut(de la începerea execuției programului respectiv)?

- a. FIFO
- b. LRU
- c. MFU
- d. NRU
- e. LFU

69. Se consideră un disc hard cu următoarele caracteristici: timpul de căutare este direct proporțional cu distanța parcursă de brațul cu capetele de citire/scriere, fiind de 2 ms între oricare doi cilindri consecutivi; timpul de transfer este neglijabil; iar latența de rotație este, în medie, de 2 ms pentru orice cerere de servire. Sistemul utilizează algoritmul de planificare SSTF. Brațul discului este poziționat inițial (i.e., la momentul $t=0$) la cilindrul 45. La momentul $t=0$ în coada de servire

ajung următoarele cereri de acces la cilindrii discului: 10 și 70, la momentul $t=50$ ms încoada de servire mai ajung și următoarele cereri de acces la cilindrii discului: 65 și 105, iar la momentul $t=125$ ms în coada de servire mai ajung și următoarele cereri de acces la cilindrii discului: 40 și 120, fiecare dintre aceste cereri provenind de la un proces distinct. Planificați servirea acceselor la disc, răspunzând la întrebările care urmează mai jos.

69.1. Indicați ordinea satisfacerii celor 6 cereri de acces la disc.

| | 1 | 2 | 3 | 4 | 5 | 6 |
|-----------------------------------|---|---|---|---|---|---|
| cererea de acces la cilindrul 10 | | | | | | |
| cererea de acces la cilindrul 40 | | | | | | |
| cererea de acces la cilindrul 65 | | | | | | |
| cererea de acces la cilindrul 70 | | | | | | |
| cererea de acces la cilindrul 105 | | | | | | |
| cererea de acces la cilindrul 120 | | | | | | |

69.2. La ce moment de timp începe servirea cererii cilindrilor 40?

Răspuns: 338

69.3. La ce moment de timp începe servirea cererii cilindrilor 120?

Răspuns: 176

69.4. Care este timpul total necesar pentru satisfacerea tuturor celor 6 cereri?

Răspuns: 402

70. Să considerăm un sistem cu paginare la cerere, cu strategia de swapping LRU, și un program ce trebuie rulat pe acest sistem, căruia SO-ul îi acordă 4 cadre (i.e., pagini fizice) pe toată durata execuției sale. Spațiul virtual al programului are 7 pagini, iar secvența de acces la ele pe parcursul execuției sale este indicată mai jos. Răspundeți la întrebările care urmează mai jos.

A1 (primul acces) : pagina 1 ,

A2 (al doilea acces) : pagina 2 ,

A3 : pagina 5 ,

A4 : pagina 4 ,

A5 : pagina 1 ,

A6 : pagina 5 ,
A7 : pagina 3 ,
A8 : pagina 4 ,
A9 : pagina 7 ,
A10 : pagina 4 ,
A11 : pagina 6 ,
A12 (ultimul acces) : pagina 4 .

70.1.Specificați, sub formă de fracție ireductibilă (fără a folosi spații, e.g. 10/17), care este rata erorilor de pagină pentru procesul din enunț?

Raspuns:.....

70.2.Care este conținutul celor 4 cadre (pagini fizice) la momentul de timp DINAINTEA accesului A11?

| cadru | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | nemapat |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

70.3.Selectați TOATE accesesele care generează erori de pagină (i.e., PFI-uri).

- a. A1
- b. A2
- c. A3
- d. A4
- e. A5
- f. A6
- g. A7
- h. A8
- i. A9
- j. A10
- k. A11
- l. A12

70.4.Care este poziția în coada LRU a paginii virtuale 5 la momentele de timp de DUPĂ fiecare dintre accesele A1 - A12?

| Locatia in coada | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 |
|-------------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|
| 0 (capat intrare) | | | | | | | | | | | | |
| 1 | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | |
| 3 (capat iesire) | | | | | | | | | | | | |
| absent din coada | | | | | | | | | | | | |

70.5.Care este conținutul celor 4 cadre (pagini fizice) la momentul de timp DINAINTEA accesului A8?

| cadru | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | nemapat |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

70.6.Care este conținutul celor 4 cadre (pagini fizice) la momentul de timp DINAINTEA accesului A4?

| cadru | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | nemapat |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

70.7.Care este poziția în memorie a paginii virtuale 5 la momentele de timp de DUPĂ fiecare dintre accesele A1 - A12?

| cadru | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 |
|----------|----|----|----|----|----|----|----|----|----|-----|-----|-----|
| 0 | | | | | | | | | | | | |
| 1 | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | |
| 3 | | | | | | | | | | | | |
| nemapata | | | | | | | | | | | | |

70.8.Care este conținutul cozii LRU la momentul de timp DINAINTEA accesului A7?

| locatia | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | empty |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

70.9.Care este conținutul cozii LRU la momentul de timp DINAINTEA accesului A4?

| locatia | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | empty |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

70.0.Care este conținutul cozii LRU la momentul de timp DINAINTEA accesului A10?

| locatia | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | empty |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

71.Se consideră un sistem de calcul SMP cu trei procesoare, ce utilizează pentru planificare algoritmul RR având configurată cuanta = 2 unități de timp, cu o singură coadă ready (i.e., folosită în comun pentru cele trei procesoare). Ordinea de alegere a procesoarelor: la RR, dacă la un moment dat mai multe procesoare sunt idle (i.e., libere), pentru a planifica un proces ready planificatorul va alege procesorul cu cel mai mic ID dintre acestea; iar în caz contrar, dacă toate procesoarele sunt ocupate și măcar unul este preemptibil, pentru a planifica un proces ready planificatorul va alege procesorul cu cel mai mic ID dintre cele care sunt preemptibile. Cinci joburi A–E (ce nu efectuează operații I/O) așteaptă să ruleze în sistem. Timpii lor de rulare și momentele de intrare în sistem sunt specificate în tabelul de mai jos. Planificați execuția celor cinci joburi în cadrul sistemului dat, răspunzând la întrebările de mai jos.

| Job | Sosire | Serviciu |
|-----|--------|----------|
| A | 0 | 6 |
| B | 1 | 7 |
| C | 1.99 | 4 |
| D | 2.99 | 4 |
| E | 3.01 | 4 |

71.1.Pentru fiecare proces, bifați TOATE intervalele de timp în care acel proces rulează pe primul procesor CPU1.

71.4. Pentru procesul E, bifați TOATE intervalele de timp în care este păstrat în coada ready.

- a. [0,1]
- b. [1,2]
- c. [2,3]
- d. [3,4]
- e. [4,5]
- f. [5,6]
- g. [6,7]
- h. [7,8]
- i. [8,9]
- j. [9,10]

71.5. Pentru procesul B, bifați TOATE intervalele de timp în care este păstrat în coada ready.

- a. [0,1]
- b. [1,2]
- c. [2,3]
- d. [3,4]
- e. [4,5]
- f. [5,6]
- g. [6,7]
- h. [7,8]
- i. [8,9]
- j. [9,10]

71.6. Ce proces are rata de penalitate cea mai mare, respectiv cea mai mică?

| | A | B | C | D | E |
|---------------------------------|---|---|---|---|---|
| Rata de penalitate cea mai mare | | | | | |
| Rata de penalitate cea mai mica | | | | | |

72. Care dintre următorii algoritmi folosiți pentru page-swapping suferă de anomalia lui Belady?

- a. LRU
- b. NRU
- c. FIFO

- d. LFU
- e. MFU

73. Bifați TOTI algoritmi de planificare care pot suferi fenomenul de infometare.

- a. algoritmul cu prioritate nepreemptiv
- b. SJF
- c. FCFS
- d. SRTF
- e. algoritmul cu prioritate preemptiv
- f. RR

74. Bifați TOTI algoritmi de planificare care permit existența arcului orientat (i.e., tranziției) de la starea running la starea ready, în graful orientat al stărilor prin care trece un proces pe parcursul vieții sale.

- a. FCFS
- b. algoritmul cu prioritate nepreemptiv
- c. SRTF
- d. algoritmul cu prioritate preemptiv
- e. SJF
- f. RR

75. Care dintre nivelele RAID asigură o performanță mai bună a accesului de date, dar NU și o siguranță mai bună de păstrare a datelor?

- a. RAID 6
- b. RAID 3
- c. RAID 1
- d. RAID 2
- e. RAID 5
- f. RAID 4
- g. RAID 0

76. Clasificați următoarele sisteme de fișiere după modul de utilizare.

| | de uz general | de uz particularizat |
|--------|---------------|----------------------|
| ext4fs | | |
| NTFS | | |
| procfs | | |

| | | |
|-------|--|--|
| btrfs | | |
| tmpfs | | |

77.Care dintre problemele de sincronizare studiate modeleaza soblonul de cooperare CREW?

- a. problema producatori-consumatori
- b. problema cititori si scriitori
- c. problema sectiunii critice
- d. problema barbierului adormit
- e. niciuna dintre cele amintite
- f. problema cinei filozofilor

78.Care dintre urmatoorii algoritmi folositi pentru optimizarea acceselor la disc are o denumire similara cu cea a unui algoritm de planificare a procesorului?

- a. C-LOOK
- b. SCAN
- c. SSTF
- d. C-SCAN
- e. FCFS
- f. LOOK

79.Bifati TOATE obiectivele de planificare ce au un caracter calitativ, nu cantitativ.

- a. rata de servire
- b. evitarea infometarii
- c. indeplinirea termenelor limita
- d. echitate
- e. gradul de utilizare a CPU
- f. timpul de raspuns

80.Ce metoda de alocare utilizeaza sistemul de fisiere ext4fs din Linux?

- a. alocare contigua
- b. niciuna dintre metodele amintite
- c. alocare inlantuita
- d. alocare indexata

81.Cum se numeste algoritmul folosit pentru page-swapping, care selecteaza drept victima o pagina care nu a fost accesata in trecutul recent?

- a. LRU
- b. LFU
- c. MFU
- d. FIFO
- e. NRU

82.Ce tip de nucleu au urmatoarele sisteme de operare?

| | nucleu monolitic | micro- nucleu | nucleu hibrid | nucleu monolitic modular | exo- nucleu |
|-------------------|---------------------|------------------|------------------|--------------------------------|----------------|
| Windows 3.x/9x | | | | | |
| MINIX | | | | | |
| Windows NT | | | | | |
| Linux | | | | | |

83.Se consideră un disc hard cu următoarele caracteristici: timpul de căutareeste direct proporțional cu distanța parcursă de brațul cu capetele decitire/scriere, fiind de 3 ms între oricare doi cilindri consecutivi; timpul detransfer este neglijabil; iar latența de rotație este, în medie, de 3 ms pentruorice cerere de servire. Sistemul utilizează algoritmul de planificare SSTF.Brațul discului este poziționat inițial (i.e., la momentul $t=0$) la cilindrul 50. Lamomentul $t=0$ în coada de servire ajung următoarele cereri de acces la cilindriidiscului: 25 și 85, la momentul $t=150$ ms în coada de servire mai ajung șiurmătoarele cereri de acces la cilindrii discului: 60 și 115, iar la momentult=300 ms în coada de servire mai ajung și următoarele cereri de acces lacilindrii discului: 35 și 120, fiecare dintre aceste cereri provenind de la unproces distinct. Planificați servirea acceselor la disc, răspunzând la întrebărilecare urmează mai jos.

83.1.La ce moment de timp incepe servirea cererii de la cilindru 115?

Raspuns:.....

83.2.Care este timpul total necesar pentru satisfacerea tuturor celor 6 cereri?

Raspuns:.....

83.3.Indicati ordinea satisfacerii celor 6 cereri de acces a disc.

| | 1 | 2 | 3 | 4 | 5 | 6 |
|--------------|---|---|---|---|---|---|
| cilindru 120 | | | | | | |
| cilindru 35 | | | | | | |
| cilindru 25 | | | | | | |
| cilindru 60 | | | | | | |
| cilindru 85 | | | | | | |
| cilindru 115 | | | | | | |

83.4.La ce moment de timp incepe servirea cererii de la cilindrul 60?

Raspuns:.....

84.Sa considerăm un sistem cu paginare la cerere, cu strategia de swapping LRU, și un program ce trebuie rulat pe acest sistem, căruia SO-ul îi acordă 4 cadre (i.e., pagini fizice) pe toată durata execuției sale. Spațiul virtual al programului are 7 pagini, iar secvența de acces la ele pe parcursul execuției sale este indicată mai jos. Răspundeți la întrebările care urmează mai jos.

A1 (primul acces) : pagina 1 ,

A2 (al doilea acces) : pagina 3 ,

A3 : pagina 6 ,

A4 : pagina 2 ,

A5 : pagina 7 ,

A6 : pagina 6 ,

A7 : pagina 4 ,

A8 : pagina 5 ,

A9 : pagina 7 ,

A10 : pagina 3 ,

A11 : pagina 7 ,

A12 (ultimul acces) : pagina 5 .

84.1. Specificați, sub formă de fracție ireductibilă (fără a folosi spații, e.g. 10/17), care este rata erorilor de pagină pentru procesul din enunț?

Raspuns:.....

84.2. Care este conținutul celor 4 cadre (pagini fizice) la momentul de timp DINAINTEA accesului A4?

| cadru | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | nemapat |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

84.3. Selectați TOATE accesele care generează erori de pagină (i.e., PFI-uri).

- a. A1
- b. A2
- c. A3
- d. A4
- e. A5
- f. A6
- g. A7
- h. A8
- i. A9
- j. A10
- k. A11
- l. A12

84.4. Care este poziția în coada LRU a paginii virtuale 7 la momentele de timp de DUPĂ fiecare dintre accesele A1 - A12?

| Locatia in coada | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 |
|------------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|
|------------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|

| | | | | | | | | | | | | |
|-------------------|--|--|--|--|--|--|--|--|--|--|--|--|
| 0 (capat intrare) | | | | | | | | | | | | |
| 1 | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | |
| 3 (capat iesire) | | | | | | | | | | | | |
| absent din coada | | | | | | | | | | | | |

84.5.Care este conținutul celor 4 cadre (pagini fizice) la momentul de timp DINAINTEA accesului A12?

| cadru | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | nemapat |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

84.6.Care este conținutul celor 4 cadre (pagini fizice) la momentul de timp DINAINTEA accesului A6?

| cadru | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | nemapat |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

84.7.Care este poziția în memorie a paginii virtuale 3 la momentele de timp de DUPĂ fiecare dintre accesele A1 - A12?

| cadru | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 |
|----------|----|----|----|----|----|----|----|----|----|-----|-----|-----|
| 0 | | | | | | | | | | | | |
| 1 | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | |
| 3 | | | | | | | | | | | | |
| nemapata | | | | | | | | | | | | |

84.8.Care este conținutul cozii LRU la momentul de timp DINAINTEA accesului A11?

| locatia | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | empty |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

84.9.Care este conținutul cozii LRU la momentul de timp DINAINTEA accesului A3?

| locatia | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | empty |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |

84.0.Care este conținutul cozii LRU la momentul de timp DINAINTEA accesului A7?

| locatia | pag 1 | pag 2 | pag 3 | pag 4 | pag 5 | pag 6 | pag 7 | empty |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
|---------|-------|-------|-------|-------|-------|-------|-------|-------|

| | | | | | | | | | | |
|---|--|--|--|--|--|--|--|--|--|--|
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |

85.2.Pentru fiecare proces, bifați TOATE intervalele de timp în care acel proces rulează pe al doilea procesor CPU2.

| | [0,1] | [1,2] | [2,3] | [3,4] | [4,5] | [5,6] | [6,7] | [7,8] | [8,9] | [9,10] |
|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |

85.3.Pentru fiecare proces, bifați TOATE intervalele de timp în care acel proces rulează pe al treilea procesor CPU3.

| | [0,1] | [1,2] | [2,3] | [3,4] | [4,5] | [5,6] | [6,7] | [7,8] | [8,9] | [9,10] |
|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |

85.4.Pentru procesul D, bifați TOATE intervalele de timp în care este păstrat în coada ready.

- [0,1]
- [1,2]
- [2,3]

- d. [3,4]
- e. [4,5]
- f. [5,6]
- g. [6,7]
- h. [7,8]
- i. [8,9]
- j. [9,10]

85.5. Pentru procesul A, bifați TOATE intervalele de timp în care este păstrat în coada ready.

- a. [0,1]
- b. [1,2]
- c. [2,3]
- d. [3,4]
- e. [4,5]
- f. [5,6]
- g. [6,7]
- h. [7,8]
- i. [8,9]
- j. [9,10]

85.6. Ce proces are rata de penalitate cea mai mare, respectiv cea mai mică?

| | A | B | C | D | E |
|---------------------------------|---|---|---|---|---|
| Rata de penalitate cea mai mare | | | | | |
| Rata de penalitate cea mai mica | | | | | |

86. Bifați TOATE tehnicile de administrare a memoriei principale care NU suferă de fenomenul de fragmentare internă.

- a. alocarea contigua a memoriei în partiții fixe
- b.** alocarea contigua a memoriei în partiții variabile
- c.** segmentare
- d. paginarea
- e. segmentarea paginată

87. Care dintre următorii algoritmi folosiți pentru optimizarea acceselor la disc are o denumire similară cu cea a unui algoritm de planificare a procesorului?

- a. FCFS
- b. SSTF**
- c. SCAN
- d. C-SCAN
- e. LOOK
- f. C-LOOK