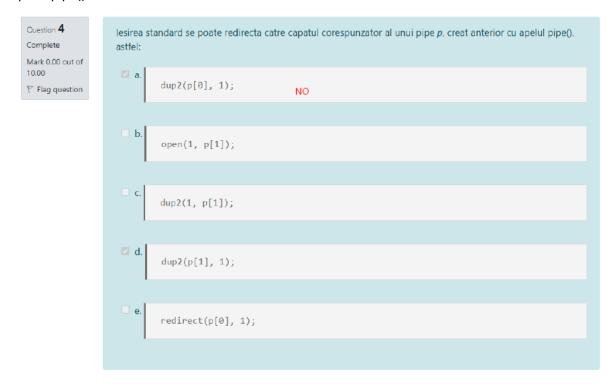
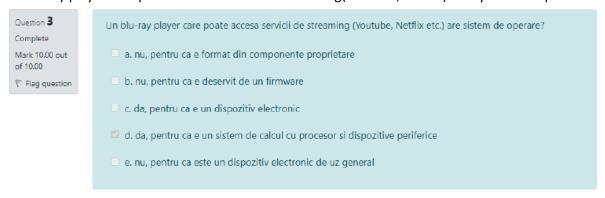
subjecteSO_eu.pdf

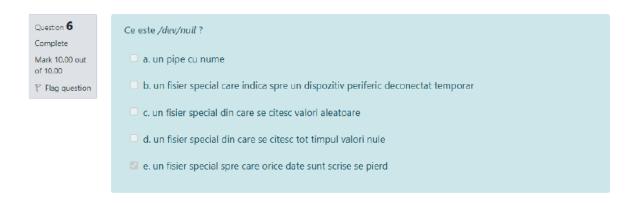
lesirea standard se poate redirecta catre capatul corespunzator al unui pipe p, create anterior cu apelul pipe() astfel:



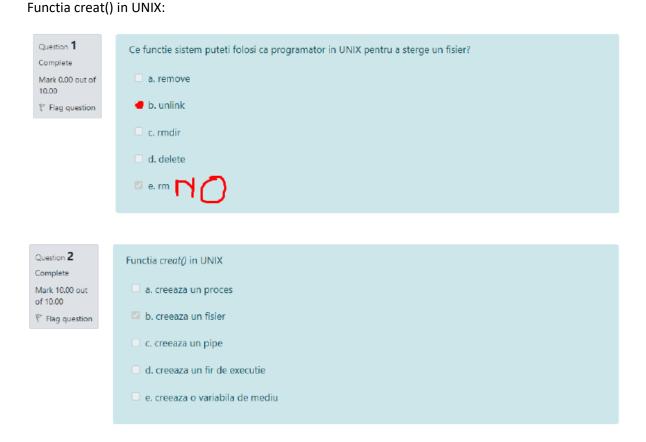
Un blu-ray player care poate accesa servicii de streaming(Youtube, Netflix) are system de operare?



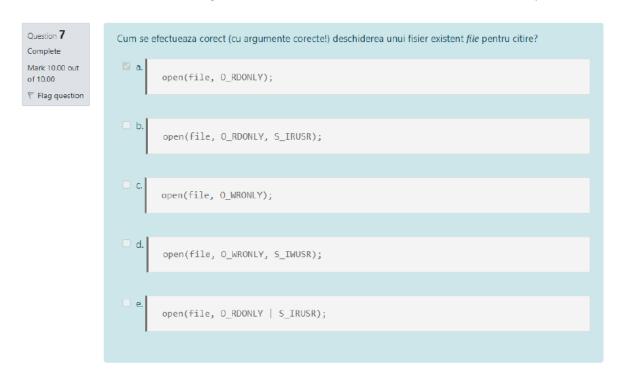
Ce este /dev/null?



Ce functie system puteti folosi ca programator in UNIX pentru a sterge un fisier?



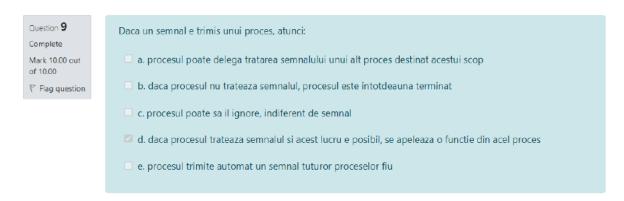
Cum se efectueaza correct (cu argumente corecte!) deschiderea unui fisier existent file pentru citire?



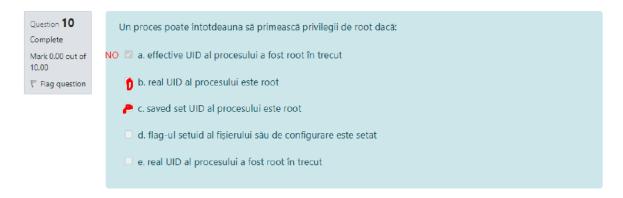
Cate procese noi creeaza urmatorul cod:



Daca un semnal e trimis unui process, atunci:



Un process poate intotdeauna sa primeasca privilegii de root, daca:



Ce se va afisa ca urmare a executiei urmatoarelor comenzi?



Daca un shell script este apelat dupa cum urmeaza:



Sub_Stefi.pdf

Numarul de blocuri de date ocupate de un fisier pe disc se poate afla:



Un programator C primeste parametrii din linia de comanda in variablila argv. Daca e compilat si apelat ca mai jos, care e argv[2]?

```
Ouestion 2
Complete
Mark 10:00 out of 10:00

F Flag question

Un program C primeste parametrii din linia de comanda in variabila argv. Daca e compilat si apelat ca mai jos, care e argv[2]?

./program "1 2" "3 1" 2 3 '1' 2 3

a. 1 2

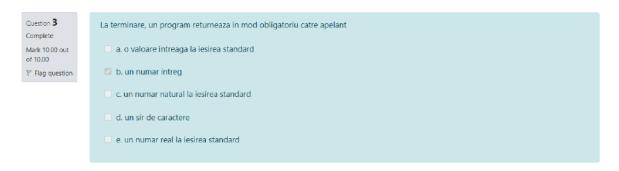
b. 3 1

c. 3

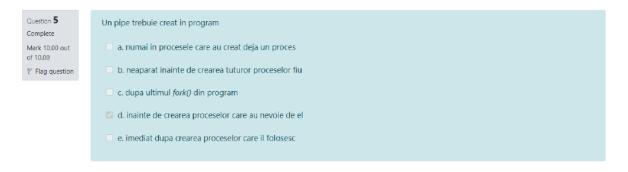
d. 1

e. 2
```

La terminare, un promgram returneaza in mod obligatoriu catre apelant



Un pipe trebuie create in program

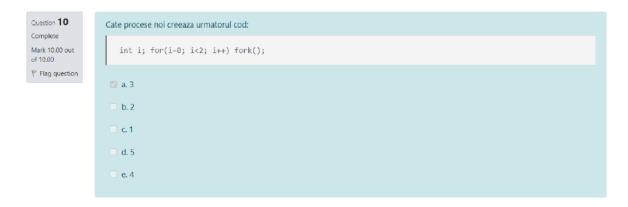


lesirea standard se poate redirecta catre un fisier, pentru care exista un descriptor valid fd, astfel:

O rutina de tratare poate



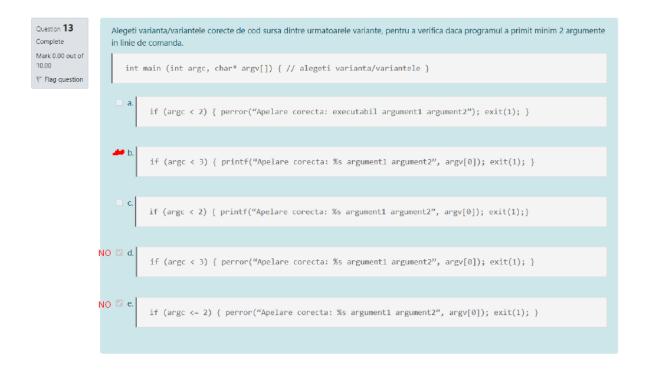
Cate procese noi creeaza urmatorul cod



Cum se efectueaza correct (cu argument corect!) crearea unui fisier file in care se va scrie continut?

Alegeti varianta/variantele corecte de cod sursa dintre urmatoarele variantele, atunci cand se doreste scrierea unui string intr-un pipe

Alegeti varianta/variantele corecte de cod sursa dintre urmatoarele variantele, pentru a verifica daca programul a primit minim 2 argumente in linie de comanda



Sub_Morariu.pdf

Exemple de sisteme de operare sunt



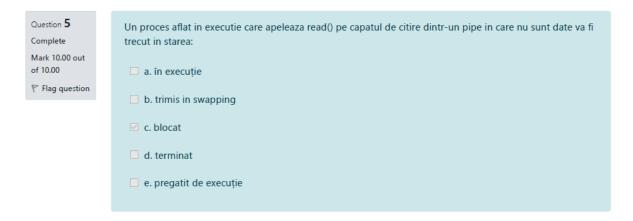
Sistemul de operare UNIX ofera, ca functionalitati specifice



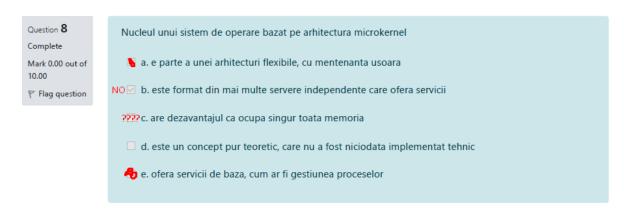
Un pipe functioneaza:



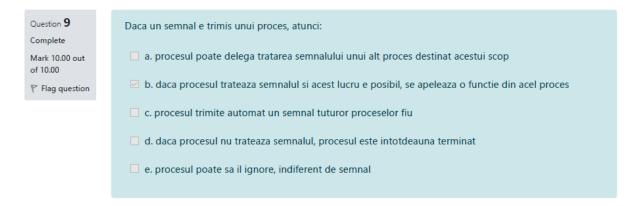
Un process aflat in executie care apeleaza read() pe capatul de citire dintr-un pipe in care nu sunt date va fi trecut in starea



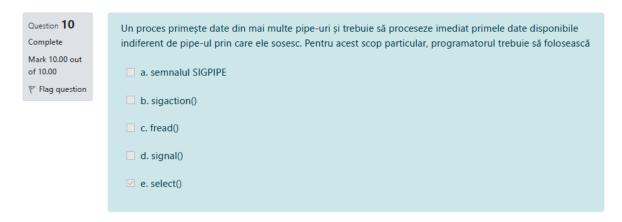
Nucleul unui system de operare bazat pe arhitectura microkernel



Daca un semnal e trimis unui process, atunci:



Un process primeste date din mai multe pipe-uri si trebuie sa proceseze imediat primele date disponibile indifferent de pipe-ul prin care eke sosesc. Pentru acest scop particular, programatorul trebuie sa foloseasca



Codul de mai jos va afisa la iesirea standard:

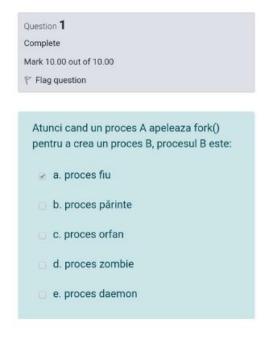
```
Question 13
Complete
Mark 0.00 out of 10.00

Frag question
```

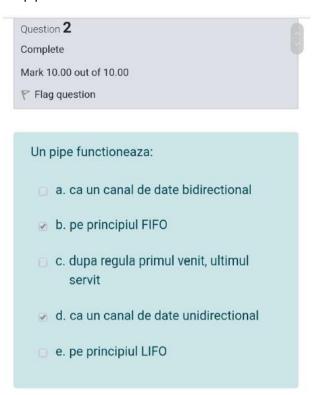
```
Codul de mai jos va afisa la iesirea standard (alegeti toate variantele corecte):
      int main(int argc, char** argv)
      {
             int pid1, pid2;
            if ((pid1 = fork()) < 0)
                     printf("Error\n");
                      exit(1);
              printf("C\n");
              if ((pid2 = fork()) < \theta)
                     printf("Error\n");
                     exit(2);
              printf("B\n");
             return 0;
     }
NO ☑ a. 3 caractere B
   ☐ b. 1 caracter B
   b c. 4 caractere B
   g d. 2 caractere C
   e. 1 caracter C
```

Sub_Sonia.pdf

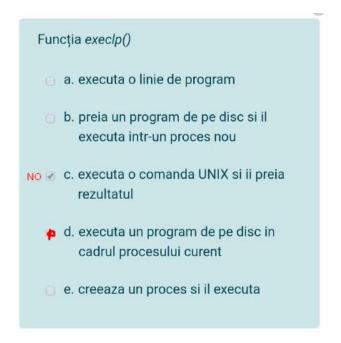
Atunci cand un proces A apeleaza fork() pentru a crea un process B, procesul B este:



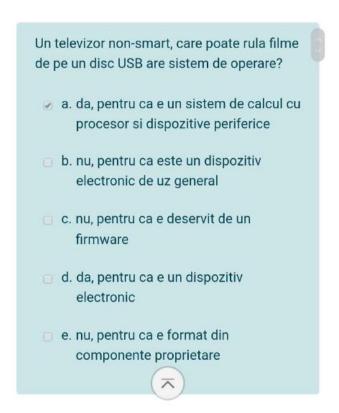
Un pipe functioneaza:



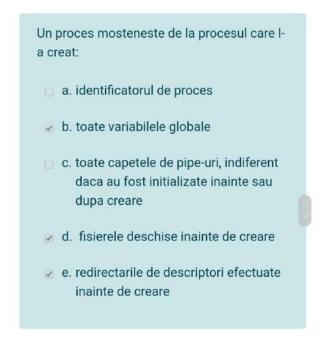
Functia execlp()



Un televizor non-smart, care poate rula filme de pe un disc USB are system de operare?

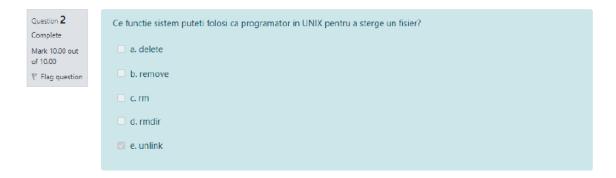


Un process mosteneste de la procesul care l-a create:

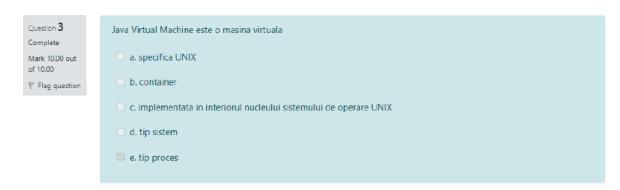


eu.pdf

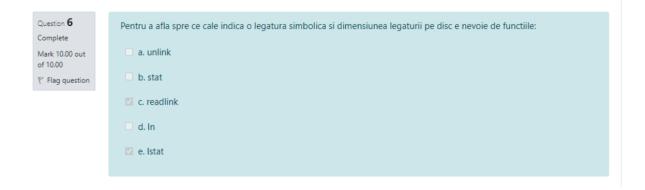
Ce functie system puteti folosi ca programator in UNIX pentru a sterge un fisier?



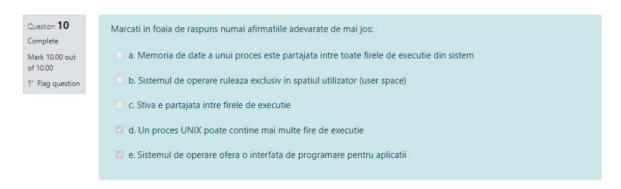
Java Virtual Machine este o masina virtuala:



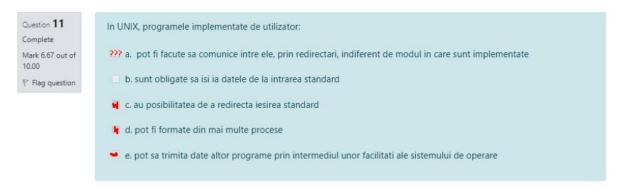
Pentru a afla spre ce cale indica o legatura simbolica si dimensiunea legaturii pe disc e nevoie de functii:



Marcati in foaia de raspuns numai afirmatiile adevarate de mai jos:

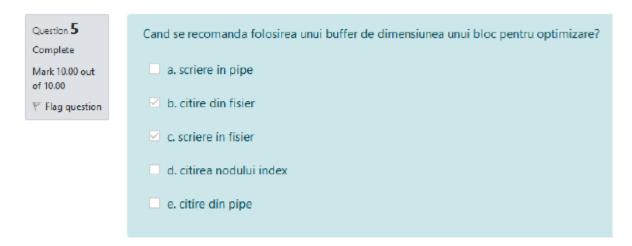


In UNIX, programele implementate de utilizator:

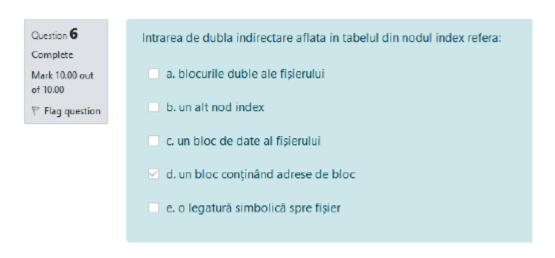


P2.pdf

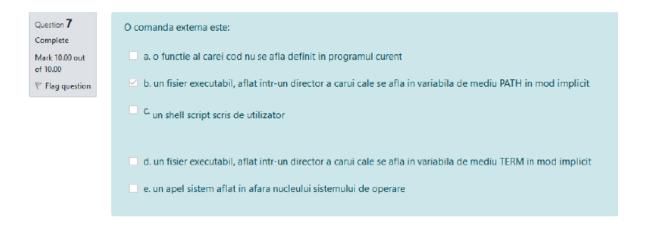
Cand se recomanda folosirea unui buffer de dimensiunea unui bloc pentru optimizarea?



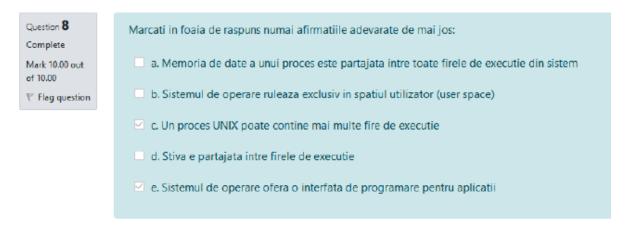
Intrarea de dubla indirectare aflata in tabelul din nodul index refera:



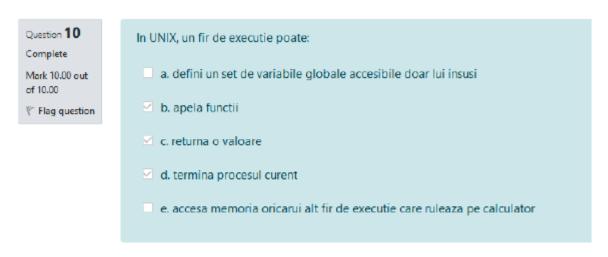
O comanda externa este:



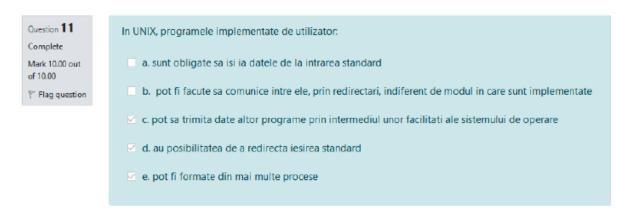
Marcati in foaia de raspuns numai afirmatii adevarate de mai jos:



In UNIX, un fir de executie poate:



In UNIX, programele implementate de utilizator:



Comanda grep primind pe cate o linie sirurile de mai jos, le va afisa la iesirea standard pe urmatoarele:

Question 13
Complete
Mark 10.00 out
of 10.00
F Flag question

```
Comanda

grep -E '^.[st].+\.t[empxt]*$'

primind pe cate o linie sirurile de mai jos, le va afisa la iesirea standard pe urmatoarele:

a. sl.text

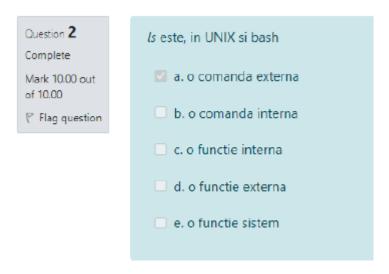
b. osl.text

c. conf.t

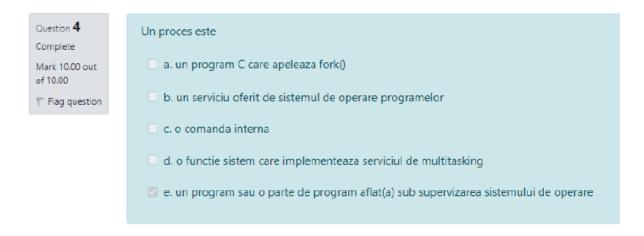
d. sl.ttm

e. config.tmp
```

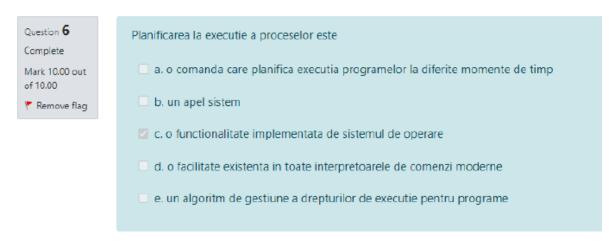
p3_2021.pdf Is este, in UNIX si bach



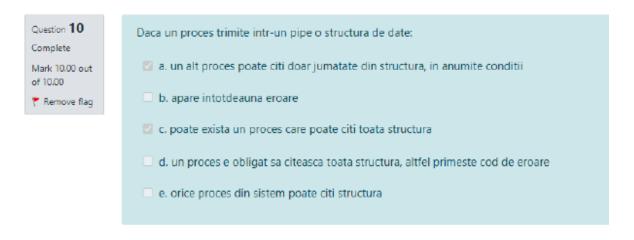
Un process este



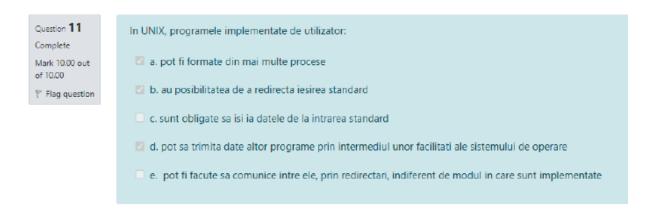
Planificarea la executie a proceselor este:



Daca un proces trimite intr-un pipe o structura de date

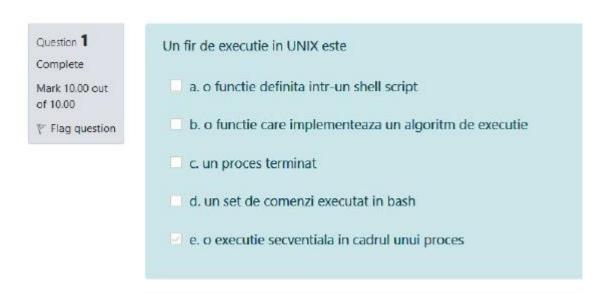


In UNIX, programele implementate de utilizator:

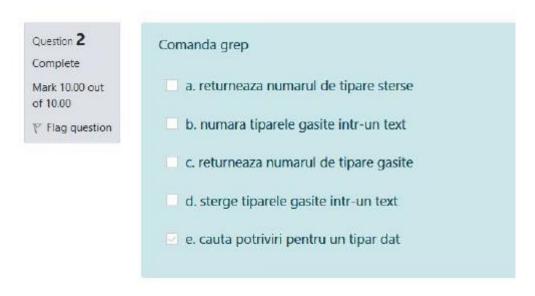


so_p2.pdf

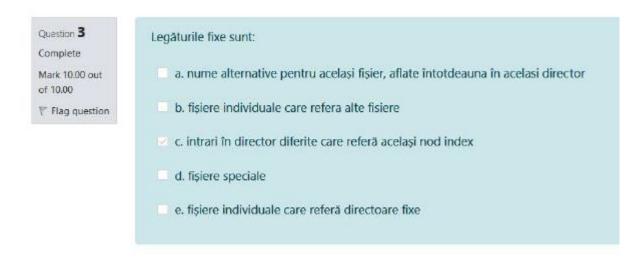
Un fir de executie in UNIX este



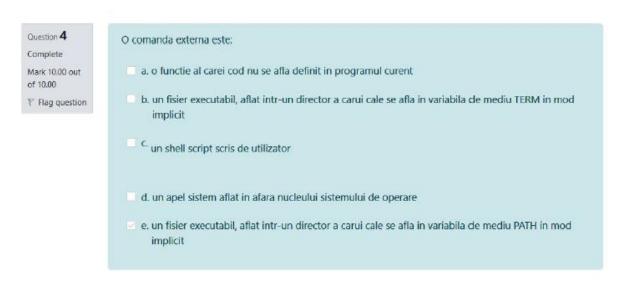
Comanda grep



Legaturile fixe sunt



O comanda externa:



Un fisier poate fi facut disponibil pentru doua procese astfel



Cand poate functia read() returna un numar mai mic de al treilea parametru primit?

Flagurile SETUID si SETGID se refera la:

Question **9**Complete

Mark 6.67 out of

Remove flag

Cand poate functia read() returna un numar mai mic decat al treilea parametru primit?

- 📥 a. cand se citeste ultimul set de date dintr-un fisier
- b. cand in fisier mai exista necititi un numar mai mare de bytes decat cel cerut
- 6 c. cand fisierul citit e mai mic, in bytes, decat numarul de octeti cerut
- 4 d. in caz de eroare
- e. cand se citesc date dintr-un pipe si procesele conectate la capatul de scriere nu mai furnizeaza date

Question 10 Complete

Mark 0.00 out of 10.00

Remove flag

Flag-urile SETUID și SETGID se referă la

- a. privilegiile de execuție din trecut ale unui proces
- b. privilegiile salvate ale unui proces aflat în execuție
- c. privilegiile de execuție actuale ale procesului
- d. drepturile de modificare a unui fișier

e. fișiere aflate în sistemul de fișiere

SUBIECTE OS - TOT CE AM

1 DOC-UL INTREB

7.0 variabila de mediu bash

1. Marcati in foaia de raspuns numai afirmatiile adevarate de mai jos:
a) Sistemul de operare are acces la spatiul nucleu(kernel space)
b) Sistemul de operare ofera o interfata de programare pentru aplicatie
c) Stiva e private fiecarui proces
d) Memoria de date a unui proces este partajata intre toate firele de executie din sistem
e) Un proces UNIX poate contine mai multe fire de executie
2. Cand poate functia read() returna un numar mai mic decat al treilea parametru primit?
a) cand se citește ultimul set de date dintr-un fișier
b) in caz de eroare
c) cand fisierul citit e mai mic, in bytes, decat numărul de octeti ceruti
3.La terminare, un program returneaza in mod obligatoriu catre apelant:
a) Un numar intreg
4.Planificarea la executie a proceselor este:
a) Un apel sistem
5.Un proces aflat in executie care apeleaza read() pe capatul de citire dintr-un pipe in care nu sunt date va fi trecut in starea:
a) blocat
6.lesirea standard se poate redirecta catre un fisier pentru care exista un descriptor valid fd, astfel:
a) dup2(fd,1)

a) e de tipul sir de caractere
8.Comanda wc -l numara
a) liniile primite la intrarea standard
9.Numarul de blocuri de date ocupate de un fisier pe disc se poate afla:
a) Parcurgând primele 10 intrari din tabela blocurilor ocupate
b) citind cele trei intrare de indirectare din tabela de blocuri
10./dev/null este
a) Un fisier special spre care orice date sunt scrise se pierd
11. Nucleul unui sistem de operare bazat pe arhitectura microkernel
a) Ofera servicii de baza, cum ar fi gestiunea proceselor
b) E parte a unei arhitecturi flexibile, cu mentenanta usoara
c) Este format din mai multe servere independente care ofera servicii
12.bash este:
a) un shell
b) un interpretor de comenzi
13.Un blu-ray player care poate accesa servicii de streaming(Youtube, Netflix, etc) are sistem de
operare?
a) da, pentru ca e un sistem de calcul cu processor si dispositive periferice
14.Un fisier poate fi facut disponibil pentru doua procese astfel:
a) un proces creeaza fisierul si apoi il creeaza pe celalalt proces
b) ambele procese cunosc descriptorul de tip int al fisierului
15.daca un proces trimite intr-un pipe o structura de date
a) orice proces din sistem poate citi structura

b) un proces e obligat sa citeasca toata structura, altfel primeste cod de eroare
16. fork() este:
a) un apel care creeaza un proces
b) o comanda care creeaza un proces nou si returneaza identificatorul sau in parinte
17. Comanda chmod 512 x va avea urmatorul efect pentru fisierul x :
a) activeaza dreptul de read pentru proprietar
b) activeaza dreptul de executie pentru grup
18. Java Virtual Machine este o masina virtuala
b) tip proces
19. informatiile despre drepturile de acces la un fisier se găsesc:
a) in nodul index aociat acestuia
b) in sistemul de operare
20. Un proces A creează un nou proces B, dupa care procesul A apeleaza cu succes functia execlp() pentru a porni un program P. Care vor fi consecintele?
a) Programul nu va mai putea prelua explicit valoarea returnata la terminare de B
b) Procesul care ruleaza P va avea același PID ca A
21.Functia kill()
a) trimite un semnal
b) termina intodeauna un proces, in mod brusc
c) distruge fara drept de apel un proces
d) capteaza un semnal
e) trimite semnalul cu nr 9 catre comanda curenta
22.Functia alarm()
a) programeaza trimiterea unui semnal anume in viitor

- 23.In UNIX, un fir de executie poate, daca este programat in acest scop
- a) termina procesul curent
- b) apela functii
- c) returna o valoare
- d) semnaleaza o situatie de eroare aparuta in executie
- e)defini un set de varabile globale accesibile doar lui insusi
- 24. Legaturile fixe sunt:
- a) intrari in director diferite care refera acelasi nod index
- 25.Un programator doreste sa transmita un sir de dimensiune variabila mai micat decat 100, prin intermediul unui pipe. Sirul este continut intr-o variabila char s[100] si a fost initializat apeland strcpy(), iar descriptorii asociati pipe-ului se afla in variabila pfd, pentru a scrie corect, in acest scop, sirul in pipe, trebuie apelat
- a) write(pfd[1],s,strlen(s));
- 25. Codul de mai jos va afisa la iesirea standard (alegeti toate variantele corecte):

```
int main(int argc, char** argv){
    int pid1,pid2;
    if((pid1 = fork()) < 0) {
        printf("Error\n");
        exit(1);
    }
    printf("C\n");
    if((pid2 = fork()) < 0) {
        printf("error\n");
        exit(2);
    }
    printf("B\n");
    return 0;
}</pre>
```

- a) 4 caractere B
- b) 2 caractere C
- c) 1 caracter B
- d) 1 caracter C
- e) 3 caractere B

26. Un program C primeste param	etrii din linia de comanda i	in variabila argv. Dacă e compilatsi
apelat ca mai jos, care e argv[2]?	./program "1 2" "3 1" 2 3	'1' 2 3

- a) 1 -> argv[1]
- b) 2 ->argv[2]
- c) 12
- d) 3 1
- e) 3 -> arg[3]

27. Un router care poate fi administrat folosind un browser Web are sist de operare?

a) da, pt ca e un system de calcul cu processor si dispositive periferice

28. Linia urmatoare:

echo "cat $f \mid wc - l$ " > g

- a) scrie intr un fisier dimensiunea (in octeti) unui alt fisier
- b) afiseaza pe ecran dimensiunea unui fisier
- c) compara nr de linii dintr un fisier cu o variabila g
- d) scrie intr un fisier nr de linii din alt fisier
- e) scrie intr un fisier sirul de caractere cat f | wc -l

29. Comanda grep -E '^.[st].+\.t[empxt]*\$' primind pe cate o linie sirurile de mai jos le va afisa la iesirea standard pe urmatoarele:

- a) conf.t
- b) os1.text
- c) config.tmp
- d) s1.ttm
- e) s1.text

30. Comanda grep 'a.[o-s].\$' primind pe cate o linie sirurile de mai jos le va afisa la iesirea standard pe urmatoarele:

- a) axpz
- b) megafonul
- c) rtatry
- d) afon
- e) megafon

31. Codul de mai jos va afisa la stdout:

```
int main(int argc, char** argv){
    int pid1,pid2;
    if((pid1 = fork()) < 0){
        printf("Error\n");
        exit(1);
    }
    if(pid != 0)
        printf("A\n");

    printf("B\n");
    if((pid2 = fork()) < 0) {
        printf("error\n");
        exit(2);
    }
    printf("A\n");
    return 0;
}</pre>
```

- a) 3 caract B
- b) 2 caract B
- c) 2 caract A
- d) 5 caract A
- e) 1 caract B

32. Informatii despre nr de legaturi spre un fisier se gasesc:

- a) in directorul current
- b) in directorul parinte
- c) in nodul index
- d) in intrarea in director
- e) in campul st_mode

33. Java Virtual Machine este o masina virtuala:

- a) tip process
- b) tip system
- c) container
- d) implementata in interiorul nucleului sistemului de operare UNIX
- e) specifica UNIX

34. Codul unei functii de biblioteca se afla:

- a) in nucleul sistemului
- b) intr un header C??
- c) intr un fisier binary
- d) in partitia de swap
- e) in varful stivei?

35. Daca un process trimite cu success date unui alt process printr un pipe, inseamna in mod necesar ca:

- a) procesele sunt fiii aceluiasi process distinct
- b) procesele au mostenit capetele necesare ale pipe ului
- c) exista un process tert care le este stramos comun celor 2 procese
- d) procesele au inchis capete nefolosite ale pipe ului
- e) procesele nu au inchis capetele utile ale pipe ului la care au acces

36. Daca un semnal e trimis unui process, atunci:

- a) daca procesul trateaza semnalul si acest lucru e posibil, se apeleaza o functie din acel process
- b) daca procesul nu trateaza semnalul, este intotdeauna terminat
- c) procesul poate desemna un proces scris special pt a trata semanlul
- d) procesul poate sa ignore semnalul, indiferent de caz
- e) procesul trimite automat un semnal tuturor proceselor fiu

37. Diferenta dintre execlp() si execl este:

- a) execlp() specifica o cale, dar execl nu
- b) execlp() tine cont de date memorate intr o anumita variabila de mediu, spre deosebire de execl care nu fol acea variabila
- c) execlp() lanseaza un process, pe cand execl nu
- d) execlp() lanseaza un process, pe cand execl() executa un fisier de comenzi dat
- e) execlp()este nerecomandat a fi folosit , spre deosebire de execl()

38. Un process este:

- a) un prog sau o parte din program aflat(a) sub supervizarea sist de operare
- b) o comanda interna
- c) un program C care apeleaza fork()
- d) un serviciu oferit de sist de op programelor
- e) o func system care implementeaza serviciul de multitasking

39. open() este:

- a) un apel system
- b) o comanda care deschide un fisier
- c) o functie de biblioteca
- d) o func aflata in interfata de programare
- e) o comanda care e implementata in nucleul sist de operare

2 DOC-UL 447

c.	. L	ie	~+	_	$\overline{}$	ď
` `\I	III I	uР	11	$\boldsymbol{\omega}$,

1	Marcati	afirmatiile	adevarate	de	mai	ios.
т.	iviai cati	anninatine	aucvarace	ue	IIIai	ıos.

a)Sistemul de operare are acces la spatiul nucleu(kernel space)

- b)Memoria unui fir de executie este partajata intre toate procesele din acel fir
- c)Un proces UNIX poate fi format din mai multe procese

d)Sistemul de operare ofera o interfata de programare pentru aplicatii

- e)stiva e privata fiecarui proces
- 2.Cand poate functia read() returna un numar mai mic decat al treilea parametru primit?
- a)Cand se citeste ultimul set de date dintr-un fisier
- b)Cand se citesc date dintr-un pipe si procesele conectate la capatul de scriere nu mai furnizeaza date

c)In caz de eroare

d)cand in fisier mai exista necititi un numar mai mare de bytes decat cel cerut

e)cand fisierul citit e mai mic, in bytes, decat numarul de octeti ceruti

3.La terminare, un program returneaza in mod obligatoriu catre apelant:

a)Un numar intreg

- b)O valoare intreaga la iesirea standard
- c)Un numar natural la iesirea standard
- d)Un sir de caractere
- e)Un numar real la iesirea standard
- 4. Planificarea la executie a programelor este:
- a)Un apel sistem
- b)O comanda care planifica executia programelor la diferite momente de timp

c)O functionalitate oferita de sistemul de operare
d)O facilitate existenta in toate interpretoarele de comenzi moderne
e)O facilitate oferita de toate shell-urile din UNIX
f)Un algoritm de gestiune a drepturilor de executie pentru programe
5.Un proces aflat in executie care apeleaza read() pe capatul de citire dintr-un pipe in care nu sunt date va fi trecut in starea:
a) blocat
b) pregatit de executie
c) in executie
d) trimis in swapping
e) terminat
6.lesirea standard se poate redirecta catre un fisier pentru care exista un descriptor valid fd astfel:
a) dup2(fd,1)
b) open(1,fd)
c) redirect(fd,1)
d) redirect(1,fd)
e) dup2(1,fd)
7. O variabila de mediu bash:
a) e de tipul sir de caractere
b) este un numar intreg
c) este un numar natural
d) este un numar real
e) poate avea orice tip
8.Comanda wc -l numara:

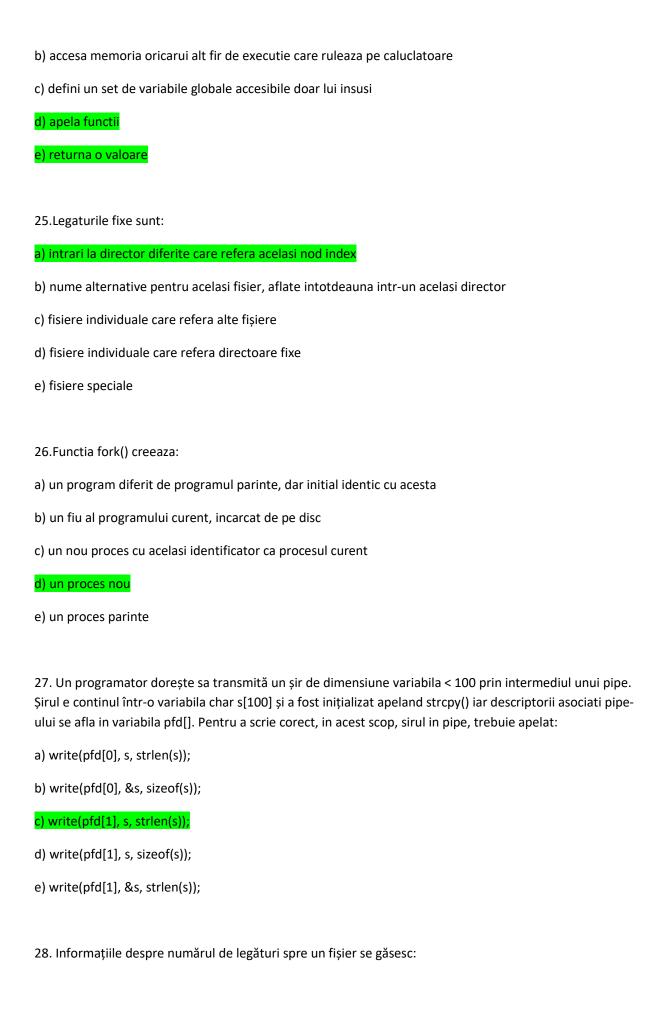
a) liniile primite la intrarea standard

b) coloanele primite la intrarea standard
c) caracterele primite la intrarea standard
d) liniile scrise la iesirea standard
e) coloanele scrise la iesirea standard
9. Numarul de blocuri de date ocupate de un fisier pe disc se poate afla:
a) parcurgand primele 10 intrari din tabela blocurilor ocupate ???
b) citind o informatie din nodul index si aplicand o formula care depinde si de dimensiunea blocului
c) citind cele 3 intrari de indirectare din tabela de blocuri
d) insumand informatiile din tabela de blocuri ocupate, aflata in nodul index
e) afland adresa pe disc ultimului bloc ocupat de fisier
10. /dev/null este:
a) un pipe cu nume
b) un fisier special din care se citesc tot timpul valori nule
c) un fisier special din care se citesc valori aleatoare
d) un fisier special spre care orice date sunt scrise se pierd
e) un fisier special care indica spre un dispozitiv periferic deconectat temporar
11. Nucleul unui sistem de operare bazat pe arhitectura microkernel:
a) este format din mai multe servere independente ce ofera servicii
b) ofera servicii de baza, cum ar fi gestiunea proceselor
c) are dezavantajul ca ocupa singur toata memoria ?
d) e parte a unei arhitecturi flexibile, cu mentenanta usoara
e) este un concept pur teoretic
12. bash este :
<mark>a) un shell</mark>

h) un chall carint
b) un shell-script
c) un apel sistem
d) un interpretor de comenzi
e) o functie de blblioteca
13. Un blu-ray player care poate accesa servicii de streaming (ex youtube,netflix,etc) are sistem de operare?
a) Da, pentru ca e un dispozitiv electronic
b) Nu, pentru ca e deservit de un filmware
c) Nu, pentru ca e un dispozitiv electronic de uz general
d) Da, pentru ca e un sitem de calcul cu procesor si dispozitive periferice
e) Nu, pentru ca e format din componente proprietare
14. Un fisier poate fi facut disponibil pentru 2 procese astfel :
a) Procesul care deschide fisierul trimite descriptorul de fisier obtinut la open() celuilalt proces
b) Un proces trimite catre celalalt numele fisierului
c) Un proces creeaza fisierul apoi il creaza pe celalalt proces
d) Se foloseste functia fdopen() care trimite un indicator de tip FILE* al fisierului
e) Ambele procese cunosc descriptorul de tip INT al fisierului
15.Daca un proces trimite intr-un pipe o structura de date
a) Un proces e obligat sa citeasca toata strctura, altfel primeste cod de eroare
b) Un alt proces poate citi doar jumatate din structura in anumite conditii ??
c) Poate exista un proces care poate citi toata structura
d) Orice proces din sistem poate citi structura
e) Apare intotdeauna eroare
16.Marcati afirmatiile adevarate
a) Sistemul de operare ruleaza exclusiv in spatiul utilizator (User space)

b) Memoria de date a unui proces e partajata intre toate firele de executie in sistem
c) Un proces UNIX poate contine mai multe fire de executie
d) Sistemul de operare ofera o interfata de programare pentru aplicatii
e) stiva e partajata intre firele de executie
17. fork() este :
a) o comanda care creeaza 2 procese
b) o functie care creeaza 2 procese distincte
c) un apel care creeaza un proces
d) un apel care creeaza un proces fiu si unul parinte
e) o comanda care creeaza un proces nou si returneaza identificatorul sau in parinte
18. Comanda chmod 512 x va avea urmatorul efect pentru fisierul x:
a) activeaza dreptul de read pentru proprietar
b) activeaza dreptul de write pentru proprietar
c) activeaza dreptul de read pentru grup
d) activeaza dreptul de executie pentru grup
e) activeaza dreptul de executie pentru ceilalti utilizatori
19. Java Virtual Machine e o masina virtuala:
a) tip sistem
b) tip proces
c) container
d) implementata in interiorul nucleului sistemulu de operare UNIX
e) e specifica UNIX
20.Informatiile despre drepturile de acces se gasesc:
a) in nodul index asociat acestuia

b) in fisier
c) in directorul care contine fisierul
d) in sistemul de operare
e) in legatura simbolica
21. Un proces A creeaza un nou proces B dupa care procesul A apeleaza cu succes functia execlp() pentru a porni un program P. Care vor fi consecintele?
a) procesul A va fi terminat complet si un nou proces va rula un program de pe disc
???b) programul nu va mai putea prelua explicit valoarea returnata la terminare de B?????????
c) procesul care ruleaza P va avea acelasi pid ca A
d) procesul care ruleaza P va avea alt pid decat A
e) B se termina automat pentru ca nu mai are parinte
22. Functia kill():
a) termina intotdeauna un proces in mod binar
b) trimite un semnal
c) distruge fara drept de apel un proces
d) capteaza un semnal
e) trimite semnalul cu numarul 9 catre comada curenta
23. Functia alarm():
a) programeaza trimiterea unui semnal anume in viitor
b) semnaleaza o situatie de eroare aparuta la executie
c) reseteaza o alarma declansata
d) informeaza un proces despre aparitia unei erori in alt proces
e) anunta procesele ca trebuie sa se termine imediat
24. In UNIX un fir de executie poate, daca este programat in acest scop :
a) termina procesul curent



a) in directorul curent
b) in directorul parinte
c) in nodul index
d) la intrarea in director
e) in campul st_mode
29. Un program C primește parametrii din linia de comanda in variabila argv. Daca e compilat si apelat ca mai jos, care e argv[2] ?
-program "1 2" "3 1" "2 3" "1" 2 3
a) 1
b) 2
c) 1 2
d) 3 1
e) 3
30.The code for a library function resides:
a. In the OS kernel
b. In a C header
c. In a binary file
d. In the swap partition
e. At the top of the stack
31.A pipe works:
a. Based on the FIFO principle
b. Based on a LIFO principle
c. Based on the FILO principle
d. As a bidirectional data channel
e. As a unidirectional data channel

32. The target of a double indirection entry from the specific data block list stored in the i-node is :
a. Another i-node
b. A block with pointers to the other blocks
c. A data block of the file
d. Double file(blocks/i-nodes)
e. A symbolik link to the file
33.A pipe must be created in a program:
a. Mandatory before creating all the processes
b. Before creating all the processes that need it
c. After the last fork() in the program
d. Immediately after creating the processes that use it
e. Only in processes that have already created a process
34.The UNIX OS provides:
a. Database management
b. Memory management
c. Application programming interface
d. Process management
Poate fi si database!!!
Management???????????????????????????????????
35. What command can we use in UNIX to create a folder:
a. Mkf
<mark>b. Mkdir</mark>
c. Newdir
d. Mdir
e. Mkfolder

36.In UNIX a thread is able to

a. terminate the current process
b. Access the memory of all the threads running on the same computer
c. Define a set of global accessible only to itself
d. Call functions
e. Return a value
37. Examples of OS:
<mark>a.Windows</mark>
b.MS-DOS
c.IBM
<mark>d.Linux</mark>
38.In UNIX, user written programs:
a. Must get their data from the standard input
b. May send data to other programs provided by the OS
c. Can be interconnected using regardless of the way they are implemented ?????????
d. Can redirect the standard output themselvs
e.Can be made of several processes
39. The kernels of an OS based on the microkernel arhitecture :
a. Is made several independent servers that offer services
b. Offers basic services such as process management
c. Has the downside that it fills the entire memory of its own

d. Is part of a flexible architecture with easy maintenance

e. Is a purely theoretical concept that has never been implemented

40. The link count of a file is found in:
a. The working directory
b. The parent directory
c. The inode
d. The directory entry
e. The st_mode field
41. Sending the strings in the answer below , each on a separate line, to the command:
grep -E '[sf].+\.t[empxt]*\$'
Will have the effect of printing at the standard output:
a. Conf.t
b. S1.txt
c. Config.tmp
d. S1.ttm
e. os1.text
42. Sending the strings in the answer below, each on a separate line, to the command:
grep -E '^[st].+\.t[empxt]*\$'
Will have the effect of printing at the standard output:
a. conf.t
b. os1.text
c. config.tmp
d. s1.ttm
e. s1.text
43. Un router care poate fi administrat folosind un browser web are system de operare?
a. da. pentru ca este un dispozitiv electronic

b. da, pentru ca e un system de calcul cu processor si dispozitive periferice

```
c. nu, pentru ca e deservit d eun firmware
d.nu, pentru ca e un dispozitiv electronic de uz general
e. nu, pentru ca e format din component proprietare
44. Comanda: echo "cat f\ | wc-l ">g
a. scrie in fisier dimensiunea (in octeti) anui alt fisier
b. afiseaza pe ecran dimensiunea unui fisier
c. compara numarul de linii dintr-un fisier cu o variabila g
d. scrie intr-un fisier numarul de linii al unui alt fisier
e. scrie intr-un fisier sirul de caractere cat f|wc-l
45. Codul de mai jos va afisa la iesirea standard:
Int main(int argc, char **argv){ int pid1,pid2;
If(pid1=fork()<0){ printf("Error\n"); exit(1);}</pre>
Printf("c\n");
If(pid2=fork()<0){ printf("Error\n"); exit(2);}</pre>
Printf("b\n");
Return 0;
}
    a. 4 caractere B
    b. 1 caracter B
    c. 2 caractere C
    d. 1 caracter C
    e. 3 caractere B
```

- 46. Un televizor non-smart care poate rula filme de pe un disc USB are sistem de operare?
- a. da, pentru ca este un dispozitiv electronic
- b. da, pentru ca este un sistem de calcul cu dispozitive periferice

- c. nu, pentru ca e deservit de un firmware d. nu, pentru ca este un dispozitiv electronic de uz gneral e. nu, pentru ca e format din componente proprietare 47. Daca un proces a trimis cu succes date unui alt proces printr-un pipe, inseamna in mod necesar ca: a. procesele sunt fiii aceluiasi proces distinct b. procesele au mostenit capetele necesare pipe-ului c. exista un proces tert care le este stramos comun celor 2 procese d. procesele au inchis capetele nefolosite ale pipe-ului e. procesele nu au inchis capetele utile ale pipe-ului la care au acces 48. Daca un semnal e trimis unui proces, atunci: a. daca procesul trateaza semnalul si acest lucru e posibil, se apeleaza o functie din acel proces b. daca procesul nu trateaza semnalul este intotdeauna terminat c. procesul poate desemna un alt proces scris pentru a trata semnalul d. procesul poate sa ignore semnalul indiferent de caz e. procesul trimite automat un semnal proceselor fiu 49. Diferenta dintre execlp() si execl() este: a. execlp specifica o cale iar execl nu b. execlp tine cont de date memorate intr-o anumita variabila de mediu c. execlp lanseaza un proces pe cand execl nu d. execlp() lanseaza un proces, pe cand execl executa un fisier de comenzi dat e. execlp() este nerecomandat spre a fi folosit, spre deosebire de execl()
- 50. Un proces este:
- a. un program sau o parte de program aflat(a) sub supervizarea sistemului de operare
- b. o comanda interna

c. un program c care apeleaza fork() d. un serviciu oferit de sistemul de operare programelor e. o functie sistem care implementeaza serviciul de multitasking 51. open() este: a. un apel sistem b. o comanda care deschide un fisier c. o functie de biblioteca d. o functie aflata in interfata de programare UNIX e. o comanda implementata in nucleul sistemului de operare 52. Comanda grep 'a.[os].\$ 'primind pe cate o linie sirurile de caractere de mai jos le va afisa la isesirea standard pe urmatoarele: a. axpz b. megafonul c. rtatry d. afon e. megafon 53. Codul de mai jos va afisa la iesirea standard: Int main(int argc, char **argv){ int pid1,pid2; If(pid1=fork()<0){ printf("Error\n"); exit(1);}</pre> Printf("B\n"); If(pid2=fork()<0){ printf("Error\n"); exit(2);}</pre> Printf("A\n"); Return 0; } a. 3 caractere B b. 2 caracter B

- c. 2 caractere A
- d. 5 caractere A
- e. 1 caracter B
- 54. Un programator doreste sa transmita un sir de dimensiune variabila mai mica decat 100, prin intermediul unui pipe. Sirul este continut intr-o variabila s[100] si a fost initializat apeland strcpy(), iar descriptorii asociati pipe-ului se afla in variabila pfd. Pentru a scrie, in acest scop, scirul in pipe, trebuie apelat:
- a. write(pfd[0],s, strlen(s))
- b. write(pfd[0], &s, sizeof(s))
- c. write(pfd[1],s, strlen(s))
- d. write(pfd[1],s, sizeof(s))
- e. write(pfd[1], &s, strlen(s))

```
15. Given the following code:
                                                                                                                                                                        int rec(char *home){
DIR* d;
struct dirent* entry;
struct stat buf;
int count= 0;
char path[100];
55.
                                                                                                                                                                           if (( d= opendir(home)) == NULL){
  printf("Error opendir on %s\n", home);
  exit(1);
}
                                                                                                                                                                              \label{eq:while} while ((entry = readdir(d)) != NULL) (if (strcmp(entry \rightarrow d_name, ".") == 0) | strcmp(entry \rightarrow d_name, ".") == 0) continue; (extrapolation of the structure) | continue; (extrapolation 
                                                                                                                                                                              strcpy(path, home);
strcat(path, "\");
strcat(path, entry→d_name);
                                                                                                                                                                              if(stat(path, &buf) <0) {
printf("Error stat on %s\n, path);
                                                                                                                                                                                             printf( t
exit(2);
                                                                                                                                                                              if( S_ISDIR(buf, st_mode)
  count += rec(path);
                                                                                                                                                                        if( $\int_{\colony, count += rec(patn),} else{
    if(S_ISREG(buf, st_mode)) count++;
}
                                                                                                                                                                              closedir(d):
                                                                                                                                                                              return count;
```

a) The regular files from dir_path are counted;
b) Execution error (" Error stat on..." because file is not found, e.g. (No such file or directory))
c) Those symbolic links from dir_path that refer to regular files are counted d) Infinite loop e) The subdirectories from dir_path are counted.

a) Si c)

56. x=10

Ce va afisa x\$x\$?

X10\$

57. Ce combinatie de comenzi va schimba directorul curent?
<mark>cc</mark>
58. Ce face functia signal()?
Specifica pentru procesul curent modul de tratare a unui semnal :P
59. Functie cu read() de citit dintr-un pipe
60. Cate procese vor fi create?
For(i=1;i <n;i++)< td=""></n;i++)<>
{fork();}
2^n-1
61. Symlink path, bytesize ?
<mark>a. Istat</mark>
b. stat
c. readlnk
d. In
62.The "wc -l" command counts:
a. Characters received at the standard input
b. Lines written to the standard output
c. Columns written to the standard output
d. Coulumns received at the standard input
e. Lines received at the standard input
63. var=3
Echo \$var && { var="abc"; echo "\$var";}

Echo \$var
a. 3, abc
b. 3, 3
c. 3, abc, abc
d. 3, abc, 3
64. A process reads from multiple pipes and always has to process the first available data in any of the pipes. For this the programmer must use:
a. sigaction()
b. The SIGPIPE signal
c. fread()
d. <mark>select()</mark>
e. signal()
65. The signal() function:
a. Signals an error
b. Handles a signal
c. Sets the behavior for handling a signal
d. Signals a process to end
e. Sends a signal
66. Under what circumstances is the return value of read() function lower that its third parameter?
a. When the last part of a file is being read
b. <mark>In case of an error</mark>
c. When the file size (in bytes) is lower than the requested byte count
d. When data is read from a pipe and the processes that use the write end of the pipe no longer send data
e. When there are more unread bytes in the file than the requested byte count

67. Th	e standard out can be redirected to a file for which a valid descriptor fd exists, as follows:			
a.	open(1, fd)			
b.	redirect(fd, 1)			
c.	dup2(1, fd)			
d.	dup2(fd, 1)			
e.	redirect(1, fd)			
68. Wł	nen having one process A that creates another process B by calling fork(), process B is			
a.	Child process			
b.	Daemon process			
c.	Zombie process			
d.	Parent process			
e.	Orphan process			
69. If a signal is sent to a process, then:				
a.	The process may assign a process written specifically to handle the signal			
b.	The process can ignore it, regardless of the signal			
c.	If the process does not handle the signal, the process is always terminated			
d.	The process automatically sends a signal to all child processes			
e.	If the process handles the signal and it is possible to do this, then a function in that process is			
	<mark>called</mark>			
70. var				
	`expr \$var + 3`'			
Echo "	`expr \$var + 3`"			
a.	26, 'expr \$var + 3`			
b.	26, 26			
C.	`expr \$var + 3`, `expr \$var + 3`			
d.	`expr \$var + 3`, 26			

71.	The	correct calls(s) - with valid arguments only!- to create a file to later write in it is			
	a.	creat(file, O_WRONLY O_CREAT O_TRUNC)			
	b.	open(file, O_WRONLY O_CREAT O_TRUNC S_IRUSR S_IWUSR)			
	c.	creat(file, S_IRUSR S_IWUSR)			
	d.	open(file, O_WRONLY O_CREAT O_TRUNC)			
	e.	open(file, O_WRONLY O_CREAT O_TRUNC, S_IRUSR S_IWUSR)			
72. Examples of operating systems are:					
	a.	MS-DOS			
	b.	UNIX			
	c.	<mark>Tizen</mark>			
	d.	Apple			
	e.	Java			
73.	Sele	ect only the true statements			
	a.	The stack is shared among threads			
	b.	The operating system offers an API			
	c.	A UNIX process may contain several threads			
	d.	The data segment of a process is shared between all the threads in the system			
	e.	The operating system runs exclusively in the user space			
74.	The	target of a double indirection entry from the specific data block list stored stored in the inode is			
	a.	Another inode			
	b.	A block with pointers to other blocks			
	c.	A data block of the file			
	d.	Double file blocks			
	e.	A symbolic link to the file			

75. Is is, in UNIX and bash

- a. An external command
- b. An internal command
- c. A system call
- d. An internal function
- e. An external function

76. A process can always gain root privileges if:

- a. The real UID of the process was root in the past
- b. The effective UID was root in the past
- c. The real UID for the process is root
- d. The setuid flag of its configuration file is set
- e. Its saved set UID is root
- 77. A child process inherits from its parent:
- a. all the values on the stack as they were before the process was created
- b. the process identifier
- c. all global variables
- d. all file descriptors redirections
- e. all open files

3 DOC-UL GRILE SO

- 1. Java Virtual Machine este o masina virtuala:
 - a. Tip sistem
 - b. Tip proces
 - c. Container
 - d. Implementata in interiorul nucleului sistemului de operare UNIX
 - e. Specifica UNIX

- 2. Informatiile despre drepturile de acces la un fisiser se gasesc:
 - a. In nodul index asociat acestuia
 - b. In fisier
 - c. In directorul care contine fisierul
 - d. In sistemul de operare
 - e. In legatura simbolica
- 3. Un proces A creeeaza un nou proces B dupa care procesul A apeleaza cu succes functia execlp() pentru a porni un program P. care vor fi consecintele?
 - a. Procesul a va fi terminat complet si un nou proces va rula un program de pe disc
 - b. Programul nu va mai putea prelua explicit valoarea returnata la terminare de B (nu prea inteleg ce scrie, sper ca e corect)
 - c. Procesul care ruleaza P va avea acelasi PID ca A
 - d. Procesul care ruleaza P va avea alt PID decat A
 - e. Se termina automt pentru ca nu mai are parinte
- 4. Functia kill()
 - a. Termina intotdeauna un procesc, in mod brusc
 - b. Transmite un semnal
 - c. Distruge fara drept de apel un proces
 - d. Capteaza (maybe) un semnal
 - e. Trimite semnalul cu numarul 9 catre comanda curenta (cred ca asta scrie)
- 5. Functia alarm()
 - a. Programeaza trimiterea unui semnal anume in viitor
 - b. Semnaleaza o situatie de eroare aparuta in executie
 - c. Ruleaza o alarma declansata
 - d. Informeaza un proces despre aparitia unei erori la alt proces
 - e. Anunta procesele ca trebuie sa se termine imediat
 - f. Reseteaza orice alarma setata anterior in acelasi process
- 6. In UNIX, un fir de executie poate, daca este programat in acest scop,
 - a. Termina procesul curent
 - b. Accesa memoria oricarui alt fir de executie care ruleaza pe calculator
 - c. Defini un set de variabile globale accesibile doar lui insuis
 - d. Apela functii
 - e. Memora o valoare
 - f. Accesa si modifica valorile variabilelor globale ale procesului in care fiul a fost creat
 - g. Accesa variabile declarate pe stiva in functiile asociate altor fire de executie
 - h. Returna o valoare
 - i. Exista in afara unui process
- 7. Legaturile fixe sunt:
 - a. Nu vad...... Maybe intrari la director diferite care refera acelasi nod index
 - b. Nume alternative pentru acelasi, aflate intotdeauna intr-un acelasi director
 - c. Fisisere individuale care referera alte fisiere

- d. Fisisere individuale care refera directoare fixe
- e. Fisierele speciale
- 8. Functia fork() creeaza
 - a. Un program diferit de programul parinte, dar initial identic cu cacesta
 - b. Un fiu al programului curent incarcat de pe disc
 - c. Un nou proces cu acelasi identificator ca procesul curent
 - d. Un proces nou
 - e. Un proces parinte
- 9. Un programator doreste sa trasnmita un sir de dimensiune variabila mai mica decar 100, prin intermediul unui pipe. Sirul este continut intr-o variabila char s[100] si a fost initializat apeland strcpy(), iar descriptorii asociati pipe-ului se afla in variabila *pfd*. Pentru a scrie corect in acest scop, sirul in pipe, trebuie apelat:
 - a. Write(pfd[0],s,strlen(s));
 - b. Write(pfd[0],&s,sizeof(s));
 - c. Write(pfd[1],s,strlen(s));
 - d. Write(pfd[1],s,sizeof(s));
 - e. Write(pfd[1],&s,strlen(s));
- 10. Un router care poate fi administrat folosind un site Web are sistem de operare?
 - a. Da, pentru ca e un dispozitiv electronic
 - b. Da, pentru ca e un sistem de calcul cu procesor si dispozitice periferice
 - c. Nu, pentru ca e deservit de firmware
 - d. Nu, pentru ca e un dispozitiv electronic de uz general
 - e. Nu, pentru ca e format din componente proprietare
- 11. Informatiile despre numarul de legaturi spre un fisier se gasesc:
 - a. In directorul curent
 - b. In directorul parinte
 - c. In nodul index
 - d. In intrarea in director
 - e. In campul st_mode
- 12. . Un program C primește parametrii din linia de comandă în varaibila argv. Dacă e compilat și apelat ca mai jos, care e argv[2]?

/program "1 2" "3 1" 2 3 '1' 2 3

- a. 3
- b. 2
- c. 12
- d. 31

- 13. Comanda *grep -E "^.[st].+\.t[enpxt]*\$"* primind pe cate o linie sirurile de mai le va afisa la iesirea standard pe urmatoarele:
 - a. Conf
 - b. St.text
 - c. Config.tmp
 - d. St.dm
 - e. pst.txt (chiar nu am vazut grilele alea, nu stiu daca ams cris ce trebuie, too blurry)
- 14. Codul de mai jos va afișa la ieșirea standard (alegeți toate variantele corecte)

```
int main( int argc, char** argv[]){
        int pid1,pid2;
        if((pid1-fork())<0) {printf("Error\n"); exit(1);}
        printf("C\n");
if((pid-fork())<0) { printf("Error\n"); exit(2);}
printf("B\n");
retun 0; }</pre>
```

- a. 1 caracter B
- b. 4 caractere B
- c. 2 caractere C
- d. 1 caracter C
- e. 3caractere B
- 15. Marcati afirmatiile corecte:
 - a. Sistemul de operare are acces la spatiul nucleu (kernel space)
 - b. Memoria de date a unui fir de executie este partajata intre toate procesele din acel fir
 - c. Un proces UNIX poate fi format din mai multe procese
 - d. Sistemul de operare ofera o interfata de programare pentru aplicatii
 - e. Stiva e privata fiecarui process
- 16. Cand poate functia read() returna un numar mai mic decat al treilea parametru primit?
 - a. Cand se citeste ultimul set de date dintr-un fiiser
 - b. Cand se citesc date dintr-un pipe si procesele conectate la capatul de scriere nu mai furnizeaza date
 - c. In caz de eroare
 - d. Cand in fisier mai exista necititi un numar mai mare de bytes decat cel cerut
 - e. Cand fisierul citit e mai mic, in bytes, decat numarul de octeti ceruti
- 17. La terminare, un program returneaza in mod obligatoriu catre apelant

a. Un numar intreg

- b. O valoare intreaga la iesirea standard
- c. Un numar natural la iesirea standard
- d. Un sir de caractere
- e. Un numar real la iesirea standard
- 18. Planificarea la executie a proceselor este:
 - a. Un apel sistem
 - b. O comanda care planifica executia programelor la diferite momente de timp
 - c. O functionalitate oferita de sistemul de operare
 - d. O facilitate existenta in toate interpretoarele de comenzi moderne
 - e. O facilitate oferita de toate shell-urile din UNIX
 - f. Un algoritm de gestiune a drepturilor de executie pentru programe
- 19. Un proces aflat in executie care apeleaza read() pe capatul de citire dintr-un pipe in care nu sunt date va fi trecut in starea:
 - a. Blocat
 - b. Pregatit de executie
 - c. In executie
 - d. Trimis in swapping
 - e. Terminat
- 20. lesirea standard se poate redirecta catre un fisier pentru care exista un descriptor balid fd, astfel:
 - a. Dup2(fd,1)
 - b. Open(1,fd)
 - c. Redirect(1,fd)
 - d. Redirect(fd,1)
 - e. Dup2(1,fd)
- 21. O variabila de mediu bash
 - a. E de tipul sir de caractere
 - b. Este un numar intreg
 - c. Este un numar natural
 - d. Este un numar real
 - e. Poate avea orice tip
- 22. Comanda wc -l numara
 - a. Liniile primite la intrarea standard
 - b. Coloanele primite la intrarea standard
 - c. Caracterele primite la intrarea standard
 - d. Liniile scrise la iesirea standard

- e. Coloanele scrise la iesirea standard
- 23. Numarul de blocuri de date ocupate de un fisier pe disc se poate afla:
 - a. Parcurgand primele 10 intrari din tabela blocurilor ocupate
 - b. Citind o informatie din nodul index si aplicand o formula care depinde si de dimensiunea blocului
 - c. Citind cele trei intrari de indirectare din tabela de blocuri
 - d. Insumand informatiile din tabela de blocuri ocupate, aflate in nodul index
 - e. Afland adresa pe disc a ultimului bloc ocupat de fisier
- 24. /dev/null este
 - a. Un pipe cu nume
 - b. Un fisier special din care se citesc tot timpul valori nule
 - c. Un fisier special din care se citesc valori aleatoare
 - d. Un fisier special spre care orice date sunt scrise se pierd
 - e. Un fisier special care indica spre un dispozitiv periferic deconectat temporar
- 25. Nucleul unui sistem de operare bazat pe arhitectura microkernel
 - a. Este format din mai multe servere independente care ofera servicii
 - b. Ofera servicii de baza, cum ar fi gestiunea proceselor
 - c. Are dezavantajul ca ocupa singur toata memoria
 - d. E parte a unei arhitecturi flexibile, cu mentenanta usoara
 - e. Este un concept pur teoretic, care nu a fost niciodata implemenat ethnic
- 26. Bash este
 - a. Un shell
 - b. Un shell script
 - c. Un apel sistem
 - d. Un interpretor de comenzi
 - e. O functie de biblioteca
- 27. Un blu-ray player care poate accesa servicii de streaming (Youtube, Netflix etc.) are sistem de operare?
 - a. Da, pentru ca e un dispozitiv electronic
 - b. Nu, pentru ca e deservit de un firmware
 - c. Nu, pentru ca este un dispozitiv electronic de uz general
 - d. Da, pentru ca este un sistem de calcul cu procesor si dispozitive periferice
 - e. Nu, pentru ca e format din componente proprietare
- 28. Un fisier poate fi facut disponibil pentru doua procese astfel:

- a. Procesul care deschide fisierul trimite descriptorul de fisier obtinut la open() celuilalt proces
- b. Un proces trimite catre celalat numele fisierului
- c. Un proces creeaza fisierul si apoi il creeaza pe celalt proces
- d. Se foloseste de functia fdopen care trimite un identificator tip *FILE al fisierului
- e. Ambele procese cunosc descriptorul de tip int al fisierului
- 29. Daca un pproces trimite intr-un pipe o structura de date
 - Un proces e obligat sa citeasca toata structura, altfel primeste cod de eroare
 - b. Un alt proces poate citi doar jumatate din structura, in anumite conditii
 - c. Poate exista un proces care poate citi toata structura
 - d. Orice proces din sistem poate citi structura
 - e. Apare intotdeauna eroare
- 30. Marcati afirmatiile adevarate:
 - a. Sistemul de operare ruleaza exclusiv inn spatiul utilizatorului (user space)
 - b. Memoria de date a unui proces este partajata intre toate firele de executie intr-un sistem
 - c. Un proces UNIX poate contine mai multe fire de executie
 - d. Sistemul de operare ofera o interfata de programare pentru aplicatii
 - e. Stiva e partajata intre firele de executie
- 31. Fork() este
 - a. O comanda care creeaza doua procese
 - b. O functie care creeaza doua procese distincte
 - c. Un apel care creeaza un proces
 - d. Un apel care creeaza un proces fiu si unul parinte
 - e. O comanda care creeaza un proces nou si returneaza identificatorul sau in parinte
- 32. Numarul de blocuri de date ocupate de un fisier pe disc se poate afla:
 - a. Parcurgand primele 10 intrari din tabela blocurilor ocupate
 - b. Citind o informatie din modul index si aplicand o formula care depinde si de dimensiunea blocului
 - c. Citind cele trei intrari de indirectare din tabela de blcouri
 - d. Insumand informatiile in tabela de blocuri ocupate, aflata in nodul index
 - e. Afland adresa pe disc a ultimului bloc ocupat de fisier
- 33. Comanda chmod 512 x va avea urmatorul efect pentru fisierul x:
 - a. Activeaza dreptul de read pentru proprietar
 - b. Activeaza dreptul de write pentru proprietar
 - c. Activeaza dreptul de read pentru grup
 - d. Activeaza dreptul de executie pentru grup

- e. Activeaza dreptul de executie pentru ceilalti utilizatori
- 34. The correct call(s) withvalid argument only to create a *file* to later write in is : Apelurile corecte pentru crearea unui fisier in care se va scrie ulterior sunt:
 - a. create(file, S_IRUSR | S_IWUSR);
 - b. open(file, O WRONLY | O_CREAT | O_TRUNC | S_IRUSR | S_IWUSR);
 - c. open(file, O_WRONLY | O_CREAT | O_TRUNC);
 - d. open(file, O_ERONLY | O_CREAT | O_TRUNC, S_IRUSR | S_IWUSR);
 - e. create (file, O_WRONLY | O_CREAT | O_TRUNC);
- 35. A process can always gain root privileges if: (Select one or more)
 - a. The setuid flag of its configuration file is set
 - b. The effective UID was root in the past
 - c. Its saved aet UID is root
 - d. The real UID of the process was root in the past
 - e. The real UID for the process is root
- 36. What system call can a programmer use in UNIX to delete a file?(Select one or more)
 - a. Unlink
 - b. Rmdir
 - c. Remove
 - d. Delete
 - e. Rm
- 37. Information on access rights to a file reside in (Select one or more)
 - a. The inode of the file
 - b. The symbolic link
 - c. The file
 - d. The directory that contains the file
 - e. The operating system
- 38. Select the correct code sequences among the following options when reading when reading a text file in a buffer

```
# define BLOCK_SIZE 4096
char buffer [BLOCK_SIZE];
Int fd =open(file_name, O_RDONLY);
```

a. while ((n= read(fd, buffer, BLOCK SIZE)) > 0)

for(int
$$I = 0$$
; $I < n$; $i++$) {...}

b. while (read(fd, buffer, BLOCK_SIZE) > 0) for(int I = 0; i < strlen(buffer); i++) {...}</p>

```
c. while ((n= read(fd, buffer, BLOCK_SIZE)) > 0) {
    buffer [n]= '\0';
    for( int i = 0; I < strlen(buffer); i++) {...}
}</li>
d. while (read(fd, buffer, BLOCK_SIZE) > 0)
    for(int = 0; buffer[i]; i++) {...}
e. while((n = read(fd, buffer, BLOCK_SIZE)) > 0)
    for(int i = 0; I < sizeof(buffer); i++) {...}</li>
```

39. A process reads from multiple pipes and always has to process the first available data in any of the pipes.

For this specific purpose, the programmer must use

Select one or more:

- a. Select()
- b. Signal()
- c. Fread()
- d. Sigaction()
- e. The SIGPIPE signal
- 40. open() is

Select one or more:

- a. a command implemented in the operating system kernel
- b. a command that opens a file
- c. system call
- d. a library function
- e. a function provided by the UNIX API
- 41. The grep command
 - a. Counts the patterns in a text
 - b. Removes the patterns in a text
 - c. Finds matches for a given pattern
 - d. Returns the number of matching patterns
 - e. Returns the number of removed patterns
- 42. The kernel of an operating system based on the microkernel architecture
 - a. Offers basic services, such as process management
 - b. Has the downside that it fills the entire memory on its own
 - c. Is part of a flexible architecture, with easy maintenance
 - d. Is made if several independent servers that offer services
 - e. Is a purely theoretical concept that has never been implemented in practice

43. What characters (and how many times) will the following code display at the standard output?

```
Int main( int argc, char** argv)
        Int pid1,pid2;
        If((pid1=fork()) < 0)
          {printf("Error\n");
        Exit(1);
        }
      If(pid1 != 0)
                 Printf("A\n");
       Printf("B");
       If ((pid2 = fork()) < 0)
        {printf("Error\n");
        Exit(2);
        }
        Printf("A\n");
        Return 0;
}
a. "B" twice
```

- b. "B" once
- c. "A" five times
- d. "A" twice
- 44. A process is
 - a. An internal command
 - b. A program or part of a program under the supervision of the OS
 - c. A system function that implements the multitasking service
 - d. A C program that calls fork()
 - e. A service offered by the operating system to the programs
- 45. When having one process A that creates another process B by calling fork(), after process A ends, process B is
 - a. Daemon process
 - b. Zombie process
 - c. Parent process
 - d. Orphan process
 - e. Kernel process

- 46. Is it recommended to use block-sized buffers for optimization when
 - a. Reading from a file
 - b. Writing to a file
 - c. Writing to a pipe
 - d. Reading the index node
 - e. reading from a pipe

incepe GX9

- 47. Codul unei functii de biblioteca se afla:
 - a. In nucleul sistemului de operare
 - b. Intr-un header
 - c. Intr-un fisier binar
 - d. In partitia de swap
 - e. In varful stivei
- 48. Un proces A creeaza un nou proces B, dupa care procesul A apeleaza cu success functia execlp() pentru a porni un program P. Care vor fi consecintele?
 - a. Procesul A nu va mai putea prelua explicit valoarea returnata la terminal de B
 - b. Procesul A va fi terminat complet si un nou process var ula un program de pe disc
 - c. Procesul care ruleaza P va avea acelasi PID ca A
 - d. Procesul care ruleaza P va avea alt PID decat A
 - e. B se termina automat pentru ca nu mai are parinte
- 49. Informatia care specifica numarul de legaturi spre un fisier se gaseste:
 - a. In directorul curent
 - b. In directorul parinte
 - c. In nodul index
 - d. In intrarea in director
 - e. In campul st_mode
- 50. Diferenta intre execlp() si execl este:
 - a. Execlp() specifica o cale, dar execl nu
 - b. Tine cont de date memorate intr-o anumita variabila de mediu, spre deosebire de execl care nu foloseste acea variabila
 - c. Execlp() lanseaza un process, pe cand execl nu
 - d. Execlp()lanseaza un proces, pe cand execl() executa un fisier de comenzi dat
 - e. Execlp() este nerecomandat a fi folosit, spre deosebire de execl()
- 51. Un process este
 - a. Un program sau o parte de program aflat(a) sub supervizarea sistemului de operare
 - b. O comanda interna

- c. Un program C care apeleaza fork()
- d. Un serviciu oferit de sistemul de operare programelor
- e. O fucntie sistem care implementeaza serviciul de multitasking

52. Open() este

- a. Un apel sistem
- b. O comanda care deschide un fisier
- c. O functie de biblioteca
- d. Functie aflata in interfata de programare oferita de UNIX
- e. O comanda care este implementata in nucleul sistemului de operare
- 53. Daca un semnal e trimis unui process, atunci:
 - a. Daca procesul trateaza semnalul si acest lucru este posibil, se apeleaza o functie din acel process
 - b. Daca procesul nu trateaza semnalul, procesul este intotdeauna terminat
 - c. Procesul poate desemna un process scris specialpentru a trata semnalul
 - d. Procesul poate sa il ignore, indifferent de semnal
 - e. Procesul trimite automat un semnal tuturor proceselor fiu
- 54. Un router care poate fi administrat folosind un browser Web are sistem de operare?
 - a. Da, pentru ca e un dispozitiv electronic
 - b. Da, pentur ca e un sistem de calcul cu processor si dispositive periferice
 - c. Nu, pentru ca e deservit de un firmware
 - d. Nu, pentru ca ese un dispozitiv electronic de uz general
 - e. Nu, pentru ca e format din componente proprietare
- 55. Linia urmatoare: echo "cat f \ wc l" > g
 - a. Scrie intr-un fisier dimensiunea(in octeti) unui alt fisier
 - b. Afiseaza pe ecran dimensiunea unui fisier
 - c. Compara numarul de linii dintr-un fisier cu o variabila g
 - d. Scrie intr-un fisier numarul de linii din alt fisiser
 - e. Scrie intr-un fisier sirul de caractere cat f | wc -l
- 56. Legaturile fixe sunt:
 - a. Intrari in director diferite care refera acelasi nod index
 - b. Nume alternative pentru acelasi fisier, aflate intotdeauna intr-un acelasi director
 - c. Fisiere individuale care refera alte fisiere
 - d. Fisiere individuale care refera directoare fixe
 - e. Fisiere speciale
- 57. Planificarea la executie a proceselor este
 - a. Un apel sistem

- b. O comanda care planifica executia programelor la diferite momente de timp
- c. O functionalotate implementata de sistemul de operare
- d. O facilitate existenta in toate interpretoarele de comenzi moderne
- e. Un argoritm de gestiune a drepturilor
- 58. Un program C primeste parametrii din linia de comanda in variabila argv. Daca e compilat si apelat cat mai jos care e argv[3]?

```
./program "1 2" "3 1" "2 3" '1' 23
a. 1
b. 2
c. 1 2
d. 3 1
e. 3
```

E 2 3

- 59. Comanda grep -E '^.[st] .+\.t[empxt]*\$' primid pe cate o liniesirurile de mai jos le va afisa la iesirea standard pe urmatoarele:
 - a. Conf.t
 - b. Osl.text
 - c. Config.tmp
 - d. Sl.tm
 - e. S1.text
- 60. Codul de mai jos va afisa la iesirea standard (alegeti toate variantele corecte):

- a. 4 caractere B
- b. 1 caracter B
- c. 2 caractere C
- d. 1 caracter C
- e. 3 caractere B

4 DOC-UL ANSWERS

Answers

D60-B

 Nucleul unui sistem de operare bazat pe arhitectura microkernel A?, bd

2. Bash este

ad

3. Un blu-ray player care poate accesa servicii de streaming (eg. Netflix) are system de operare?

d

4. Un fisier poate fi facut disponibil pentru doua procese astfel ce

5. Daca un proces trimite intr-un pipe o structura de date

c

6. Afirmatii adevarate

bcd

7. Fork() este

ce

8. Nr de blocuri de date ocupate de un fisier pe disc se poate afla

ac

9. Comanda chmod 512 x va avea urmatorul effect pentru fisierul x ad