

Exerciții aplicative

Al treilea formular cu întrebări de evaluare.

[bash] Să se specifice ordinea corectă de înlănțuire a celor 4 comenzi de mai jos, astfel încât lanțul obținut în acest fel să afișeze 1 numai dacă fișierul test.so poate fi accesat în citire de către ceilalți utilizatori din grupul proprietarului, și respectiv 0, dacă nu. (1p)

	1	2	3	4
stat --format "%A" test.so	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
cut -c5-7	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
grep r	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
wc -l	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

[bash] Să se specifice ordinea corectă de înlănțuire a celor 4 comenzi de mai jos, astfel încât lanțul obținut în acest fel să afișeze toate procesele din sistem în formatul user:comanda:pid , ordonate după numele comenzii. (1p)

	1	2	3	4
ps -eo user,comm,pid --no-headers	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
tr -s " "	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
cut -d" " -f1,2,3 --output-delimiter=:	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
sort -k2 -t:	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

[bash] Să se specifice ordinea corectă de înlănțuire a celor 4 comenzi de mai jos, astfel încât lanțul obținut în acest fel să afișeze numele fișierului și dimensiunea pentru al treilea cel mai mare (ca dimensiune) fișier din directorul curent și subdirectoarele sale. (1p)

	1	2	3	4
find . -printf "%p %s\n"	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
sort -k2 -n	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
tail -n 3	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
head --lines=1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

[bash] Să se scrie comanda simplă find care afișează, pentru toate fișierele cu extensia .cpp din directorul curent (parcurs recursiv pe maxim 3 nivele), doar numele acelora în al căror conținut apare textul "define" (apelați comanda adecvată prin -exec). (2p)

Respectați formatul de output descris prin exemplul de mai jos:

./so2021/prog1.cpp

./so2021/test/director/prog2.cpp

find . -maxdepth 3 -name "*.cpp" -exec grep define {} \;

[bash] Scrieți comanda înlănțuită care selectează DOAR lista fișierelor și directoarelor aflate în directorul curent și al căror nume începe cu prefixul "lab", apoi le afișează doar pe al doilea și pe al treilea din această listă. (2p)

ls . | grep -E "^lab" | head -n 3 | tail -n 2

[C] Scrieți apelul POSIX prin care se creează, în directorul curent de lucru, un link hard, cu numele "alias.txt", către fișierul cu numele "afile.txt" aflat în directorul părinte al directorului curent de lucru. Observații: 1. NU folosiți SPAȚII. 2. Terminați apelul cu ; (PUNCT și VIRGULĂ). (0.5p)

[C] Scrieți apelul POSIX prin care se deschide fișierul "fisier.txt" atât pentru citire, cât și pentru scriere, iar dacă fișierul nu există, atunci îl și creează, cu permisiuni de acces în citire și scriere doar pentru proprietar, specificate printr-o constantă numerică în OCTAL. Observații: 1. NU folosiți SPAȚII. 2. Terminați apelul cu ; (PUNCT și VIRGULĂ). (0.5p)

open("fisier.txt",O_RDWR|O_CREAT,0700);

[C] Se consideră următoarea secvență de cod ce execută apelul POSIX stat pe fișierul script.sh. Bifați doar opțiunile ce sunt MINIM necesare astfel încât execuția codului să afișeze pe ecran textul "ABC". (1p)

```
struct stat info;  
if (stat("script.sh", &info) == 0) {  
    if (info.st_mode & S_IWUSR) printf("A");  
    if (info.st_mode & S_IRGRP) printf("B");  
    if (info.st_mode & S_IXOTH) printf("C");  
}
```

- ☐ Proprietarul fișierului are drept de citire.
- ☐ Proprietarul fișierului are drept de scriere.
- ☒ Proprietarul fișierului are drept de execuție.
- ☒ Colegii din grupul proprietar al fișierului au drept de citire.
- ☐ Colegii din grupul proprietar al fișierului au drept de scriere.
- ☐ Colegii din grupul proprietar al fișierului au drept de execuție.
- ☐ Utilizatorii care nu sunt în grupul proprietarului fișierului au drept de citire.
- ☒ Utilizatorii care nu sunt în grupul proprietarului fișierului au drept de scriere.
- ☐ Utilizatorii care nu sunt în grupul proprietarului fișierului au drept de execuție.

[C] Se consideră următoarele declarații din figura alăturată. Ce acțiune realizează apelul `fcntl(fd, F_SETLKW, &L);` ? (1p)

```
struct flock L;  
L.l_type = F_WRLCK;  
L.l_whence = SEEK_END;  
L.l_start = -2*sizeof(int);  
L.l_len = 2*sizeof(int);
```

- ☐ Blochează pentru scriere ultimii doi octeți, din fișierul referit prin descriptorul `fd`.
- ☐ Deblochează pentru scriere o zonă de dimensiunea a două numere întregi, din fișierul referit prin descriptorul `fd`.
- ☐ Blochează pentru citire o zonă de dimensiunea a două numere întregi, de la sfârșitul fișierului referit prin descriptorul `fd`.
- ☒ Blochează pentru scriere o zonă de dimensiunea a două numere întregi, de la sfârșitul fișierului referit prin descriptorul `fd`.
- ☐ Blochează pentru scriere o zonă de dimensiunea a două numere întregi, începând cu deplasamentul curent pentru fișierul referit prin descriptorul `fd`.
- ☐ Deblochează pentru citire o zonă de dimensiunea a două numere întregi, începând cu deplasamentul curent pentru fișierul referit prin descriptorul `fd`.
- ☐ Blochează pentru scriere și pentru citire o zonă de dimensiunea a doi octeți, de la deplasamentul curent pentru fișierul referit prin descriptorul `fd`.