

Test Practic Midterm USO

Varianta Model, 19 Noiembrie 2016

Nume: _____

Grupa: _____

Indicații

- Testul conține 6 subiecte. Fiecare subiect se punctează cu maxim 20 puncte pentru o rezolvare corectă și completă. Rezultatul se trunchiază la 100 de puncte.
- Se pot obține punctaje parțiale pentru un subiect numai dacă acest lucru este specificat.
- Ordinea de rezolvare a subiectelor este la alegerea voastră.
- Pentru a fi punctată, o rezolvare **trebuie** să includă și metoda de verificare a funcționalității acesteia.
- Rezolvările subiectelor pot fi realizate utilizând **doar** aplicații disponibile în linia de comandă.
- Subiectele se rezolvă pe mașinile virtuale puse la dispoziție fiecărui student. Orice rezolvare pe mașina locală este nevalidă.
- Dacă nu este menționată calea completă a unui fișier sau director, acesta trebuie creat în `/home/student`.
- Conectați-vă pe `fep.grid.pub.ro` prin SSH folosind comanda `ssh -X -o ServerAliveInterval=100 $user@fep.grid.pub.ro` unde `$user` este contul vostru de pe `cs.curs.pub.ro`.
- Folosiți comanda `startexam_uso.sh` pentru a vă conecta la mașina virtuală `router` pe care o veți folosi pe parcursul testului. Introduceți parola `student`.

Punctaj total

Subiecte

1. Faceți în așa fel încât de fiecare dată când se deschide un shell bash, utilizatorul să fie întâmpinat cu mesajul Salut, `<nume utilizator>!`. 20
Hint: Variabila USER.
2. Afișați numărul directoarelor copil ale lui / ce conțin directoare home pentru utilizatorii din sistem. 20
Exemplu: Dacă utilizatorii `razvan`, `student`, și `nobody` au directoarele home `/home/razvan`, `/home/student`, respectiv `/var/tmp`, rezultatul este 2, adică doar subdirectoarele `home` și `var` ale lui / conțin home-uri.
3. Afișați, pe două coloane, process ID-ul și CMD pentru toate procesele inițiate de root. 20
4. Creați fișierul `poly.c` care să conțină următoarea sursă C:

```
#include <stdio.h>
void main() {
    printf("USO RULZ!\n");
}
```

 - (a) Generați executabilul `poly` pornind de la sursa `poly.c`. 6
 - (b) Generați executabilul `poly.small` astfel încât să fie optimizat pentru a avea o dimensiune cât mai mică (*Hint:* `man gcc`) 8
 - (c) Afișați doar executabilele rezultate (care încep cu `poly`), ordonate în funcție de dimensiune. 6

5. (a) În directorul home al utilizatorului **student**, creați următoarea structură: 10
- ```
carne/
|-- porc
|-- pui
'-- vita
legume/
|-- cartofi
'-- rosii
```
- Notă:** directoarele **carne** și **legume** conțin fișiere, nu alte directoare.
- (b) Configurați permisiunile fișierelor astfel încât utilizatorul **andrei** să poată scrie și citi fișierele **pui**, **porc** și **carofi**, dar să nu aibă deloc acces la fișierele **rosii** și **vita**. 10  
**Hint:** Pentru a vă autentifica drept **andrei** fără a îi cunoaște parola folosiți **sudo**.
6. (a) Afișați toate fișierele din directorul **/home/student** și subdirectoarele acestuia care au fost modificate în ultimele 24 de ore. 10  
**Hint:** **find**
- (b) Pentru fișierele de mai sus, afișați toate detaliile: permisiuni, owner, data modificării, etc. 10  
**Hint:** **ls -l**