

## Test Practic Final USO

Varianta 104, 30 August 2018

Nume: \_\_\_\_\_

Grupa: \_\_\_\_\_

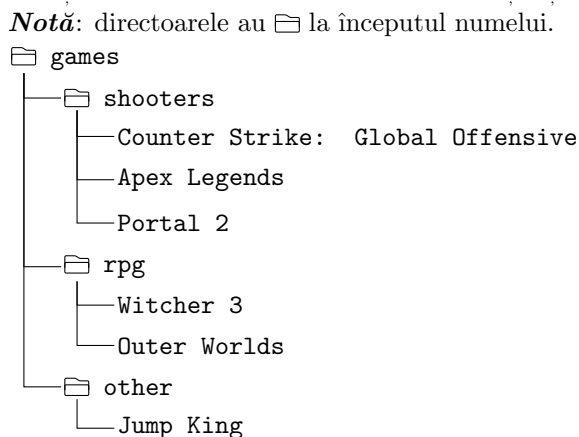
### Indicații

- Testul durează 100 minute și conține 6 subiecte. Fiecare subiect se punctează cu maxim 20 de 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 `uso` pe care o veți folosi pe parcursul testului. Introduceți parola `student`.

Punctaj total

### Subiecte

1. (a) Creați următoarea structură de directoare și fișiere:



- (b) Creați o arhivă de tipul **gzip** numită „`all_them_games.tar.gz`” conținând ierarhia de fișiere creată anterior.
- (c) Creați utilizatorii „`valve`” și „`cdpr`”.
- (d) Faceți modificările necesare astfel încât asupra directorului „`shooters`” și a tuturor fișierelor din acesta utilizatorul „`valve`” să aibă drepturi de scriere și citire, iar utilizatorul „`cdpr`” doar drepturi de scriere.
- (e) În directorul „`other`” creați un ciclu folosind 3 legături simbolice.
2. (a) Afișați numărul de cuvinte din fișierul „`/usr/include/termio.h`”.
- (b) Afișați numărul de apariții ale caracterului „`a`” din fișierul „`/etc/shadow`”.

- (c) Afișați numărul de aplicații distincte care sunt configurate ca shell implicit pentru utilizatorii din sistem. 5
  - (d) Scrieți un one-liner care afișează suma tuturor identificatorilor utilizatorilor de pe sistem. 8
- (a) Afișați toate dispozitivele de tip caracter de pe sistem. 5
  - (b) Creați un fișier de dimensiune **10 M** care să conțină **doar** litere mici ASCII aranjate în mod aleator. 6
  - (c) Formatați fișierul de **10 M** de la subpunctul anterior în format „**ext3**”. 3
  - (d) Afișați numărul celei de-a treia linii pe care se află cuvântul „**include**” din fișierul „**/usr/include/aio.h**”. 6
- (a) Scrieți un script numit „**4a.sh**” care primește ca argument o adresă și afișează adresa IP a asociată acesteia. Script-ul va întoarce codul de eroare 1 dacă va primi mai mult sau mai puțin de un argument.  
**Notă:** testați folosind adresa „**elf.cs.pub.ro**”. 5
  - (b) Modificați script-ul astfel încât acesta să afișeze și al doilea hop prin care trec pachetele în drum spre adresa respectivă. 5
  - (c) Modificați script-ul astfel încât acesta să scrie toate hop-urile în afara de al doilea în fișierul „**extra.txt**”. 5
  - (d) Modificați script-ul astfel încât acesta să trimită 3 pachete către adresa primită ca argument, la interval de 0.3 secunde. Fiecare pachet trebuie să aibă ca padding doar biți de 1. 5
- (a) Afișați de câte ori apare în fișierul „**/etc/passwd**” fiecare program care este configurat ca shell implicit. 5
  - (b) Afișați **doar** numele utilizatorilor care au ca shell implicit „**/usr/sbin/nologin**”. 5
  - (c) Pentru primele 10 procese ordonate după durata de timp de când acestea au pornit, afișați PID-ul, PPID-ul, utilizatorul și durata respectivă. 5
  - (d) Scrieți script-ul „**5d.sh**” care primește ca argument un PID și afișează PID-urile tuturor proceselor copil ale procesului identificat prin PID-ul primit ca argument. 5
- (a) Descărcați arhiva de la următorul link și dezarhivați-o:  
„**http://elf.cs.pub.ro/uso/res/final/20-ian/four.zip**”. 3
  - (b) Modificați Makefile-ul astfel încât compilarea să funcționeze. 5
  - (c) Modificați Makefile-ul astfel încât fișierele obiect să fie înaintea link-editării lor într-un executabil final. 5
  - (d) Adăugați în fișierul „**Makefile**” o regulă numită **pack** care va crea o arhivă **zip** conținând fișierele sursă și fișierul „**Makefile**”. 3
  - (e) Adăugați în fișierul „**Makefile**” o regulă numită **clean** care va șterge fișierele obținute din procesul de compilare, inclusiv fișierul creat de regula **pack**. Asigurați-vă că regula **clean** poate să fie rulată și în prezența unui fișier numit „**clean**”. 4