

Laborator #5 : exerciții de laborator

Sumar:

I) [Exerciții cu fișiere de comenzi \(partea a doua -- *script*-uri ce efectuează diverse procesări de fișiere\)](#)

- a) [Exerciții propuse spre rezolvare](#)
- b) [Exerciții suplimentare, propuse spre rezolvare pentru acasă](#)

I) **Exerciții cu fișiere de comenzi (partea a doua -- *script*-uri ce efectuează diverse procesări de fișiere) :**

a) *Exerciții propuse spre rezolvare :*

Intrați pe setul de exerciții propuse spre rezolvare, pe care vi-l va indica profesorul de laborator, în timpul laboratorului, și încercați să le rezolvați singuri:

Setul #1

Setul #2

Setul 1

1. [#1: Call2Find]
Să se scrie un script care va modifica permisiunile pentru toate fișierele de tip obișnuit ce sunt scripturi bash, aflate în directorul curent și în toate subdirectoarele acestuia, adăugându-le dreptul de execuție pentru toți utilizatorii.
Atenție: NU se cere ca parcurgerea recursivă a directorului respectiv să fie explicită, i.e. puteți folosi comanda `find` sau comanda `ls -R` pentru a parcurge directorul specificat.
(Indicație: filtrați fișierele de tip obișnuit cu comanda `find` și verificați conținutul fișierului folosind comanda `file` pentru a vedea dacă este script bash.)
2. [#2: MyFind]
Să se scrie un script care calculează și afișează numărul total de fișiere de tip obișnuit ce sunt "ascunse" (i.e., cele ale căror nume începe cu caracterul '.'), numărul total de fișiere de tip alias (numărați doar link-urile simbolice) și numărul total de subdirectoare aflate într-un director dat, parcurgand acel director în manieră recursivă, în mod explicit (i.e., fără a apela comanda `find` sau comanda `ls -R`). Numele directorului de pornire se transmite ca parametru în linia de comandă, sau se va citi prin comanda `read` în caz contrar.
(Indicație: parcurgeți cu un `for` directorul curent și calculați informațiile solicitate, iar pentru fiecare intrare din director care este de tipul (sub)director apelați recursiv scriptul.)

Show / Hide some suggestions for solving the problem

Ideea de rezolvare:
-- recursia se poate implementa în două maniere diferite: fie prin apel recursiv de funcție, fie prin apel recursiv de script;
-- statisticile (i.e. valorile ce se cer a fi calculate în enunțul problemei) se calculează într-o manieră asemănătoare cu cea din rezolvările date la exercițiul rezolvat [MyFind #1] din suportul de laborator.
Notă: citiți cu atenție și observația de la finalul rezolvărilor date pentru exercițiul rezolvat [MyFind #1] din suportul de laborator.

1. [#1: Call2Find]

Să se scrie un script care va copia toate fişierele cu extensia `.sh`, aflate într-un director primit ca parametru la linia de comandă şi în toate subdirectoarele acestuia, într-un folder numit `backup_sh`, schimbându-le numele original în hash-ul lor `sha256`.

Atenţie: NU se cere ca parcurgerea recursivă a directorului respectiv să fie explicită, i.e. puteţi folosi comanda `find` sau comanda `ls -R` pentru a parcurge directorul specificat. (Indicaţie: filtraţi şi copiaţi fişierele cu comanda `find`, iar hash-urile `sha256` le puteţi calcula cu ajutorul comenzii `sha256sum`.)

2. [#2: MyFind]

Să se scrie un script care primeşte două argumente: un nume de director şi un număr `max_depth`. Scriptul va parcurge acel director în manieră recursivă, în mod explicit (i.e. fără a apela comanda `find` sau comanda `ls -R`), până la adâncimea `max_depth`, şi va afişa *path*-ul fiecărui fişier de tip obişnuit găsit, însoţit de eticheta "Script bash", în cazul în care fişierul conţine cuvântul `/bin/bash`, respectiv însoţit de eticheta "Nu-i script bash", în caz contrar.

(Indicaţie: parcurgeţi cu un `for` directorul curent şi prelucraţi fişierele întâlnite conform enunţului, iar pentru fiecare intrare din director care este de tipul (sub)director apelaţi recursiv scriptul.)

Show / Hide some suggestions for solving the problem

Ideea de rezolvare:

- recursia se poate implementa în două maniere diferite: fie prin apel recursiv de funcţie, fie prin apel recursiv de script;
- statisticile (i.e. valorile ce se cer a fi calculate în enunţul problemei) se calculează într-o manieră asemănătoare cu cea din rezolvările date la exerciţiul rezolvat [MyFind #1] de mai sus.

Notă: citiţi cu atenţie şi observaţia de la finalul rezolvărilor date pentru exerciţiul rezolvat [MyFind #1] de mai sus.

b) *Exerciţii suplimentare, propuse spre rezolvare pentru acasă :*

Iată alte câteva exerciţii cu fişiere de comenzi, pe care să încercaţi să le rezolvaţi singuri în timpul liber, pentru a vă auto-evalua cunoştinţele dobândite în urma acestui laborator:

1. [Call2Find #3]

Să se scrie un script care "scanează" directorul primit ca parametru pentru a găsi fişiere MP3, în acesta sau prin subdirectarele lui (*Notă:* dacă scriptul nu este apelat cu parametru, atunci va folosi implicit directorul home `~`).

Pentru toate fişierele MP3 găsite scriptul va crea link-uri hard (Indicaţie: [vezi comanda `ln`](#)) plasate în directorul `~/html/muzica/`.

Scriptul va genera deasemenea un "playlist" în format html simplu, plasat în directorul `~/html`, şi care va conţine numele fişierelor MP3 găsite (Indicaţie: [eventual doar numele fără extensie, vezi comanda `basename`](#)), precum şi link-uri HTML spre acestea (Indicaţie: este suficient un link de forma `nume mp3 fara extensie`).

În plus, scriptul va verifica permisiunile pentru directoare şi fişiere în aşa fel încât în final playlistul să poată fi accesat din exterior pe web (Indicaţie: [trebuie setate drepturi de execuţie pentru toţi \(sau măcar pentru others\) pe directoarele `~`, `~/html` şi `~/html/muzica`, precum şi drept de citire pentru toţi \(sau măcar pentru others\) pe fişierul playlist creat](#)).

2. [Call2Find #4]

Să se scrie un script care să construiască o pagină web cu link-uri HTML către toate fişierele imagini (i.e., cele cu extensiile `.jpg`, `.gif`, `.png`, ş.a.) din contul personal d epe server, ordonate după data ultimei modificări a fişierelor.

(Indicaţie: [aveţi grijă la drepturi pentru vizibilitatea pe web a paginii construite, similar ca la exerciţiul precedent](#).)

3. [MyFind #4]

Să se scrie un script care calculează numărul total de cuvinte şi respectiv lungimea cea mai mare a liniilor de text, din toate fişierele ce conţin text obişnuit (i.e., fişierele pentru care comanda `file` raportează "ASCII text") aflate într-un director dat, parcurgand acel director în manieră recursivă, în mod explicit (i.e., fără a apela comanda `find` sau comanda `ls -R`). Numele directorului se transmite ca parametru în linia de comandă, sau se va citi prin comanda `read` în caz contrar.

(Indicație: parcurgeți cu un for directorul curent și calculați informațiile solicitate, iar pentru fiecare intrare din director care este de tipul (sub)director apălați recursiv scriptul.)

Cerință suplimentară: să se afișeze acele informații totale pentru fiecare subdirector parcurs.

Show / Hide some suggestions for solving the problem

Ideea de rezolvare:

-- recursia se poate implementa în două maniere diferite: fie prin apel recursiv de funcție, fie prin apel recursiv de script;

-- statisticile (i.e. valorile ce se cer a fi calculate în enunțul problemei) se calculează într-o manieră asemănătoare cu cea din rezolvările date la exercițiul rezolvat [MyFind #2] din suportul de laborator.

Notă: citiți cu atenție și observația de la finalul rezolvărilor date pentru exercițiul rezolvat [MyFind #1] din suportul de laborator.

4. [MyLast]

Să se scrie un script care citește nume de utilizatori de la intrarea standard (folosind comanda read) și, pentru fiecare utilizator introdus, va afișa ultimele 3 conectări (logări) ale acestuia (în luna curentă), iar dacă acesta nu s-a conectat deloc în luna curentă va afișa mesajul "Utilizatorul X nu s-a conectat niciodată!".

(Indicație: a se folosi comanda last.)

5. [MyZip]

Să se scrie un script care primește ca parametru un director (fie de la linia de comandă, fie de la tastatură) și arhivează toate fișierele text (i.e., cele ce au extensia .txt) și toate scripturile (i.e., fișierele ce au extensia .sh) aflate doar în acest director (deci nu și cele din subdirectoarele lui!).

Arhivarea fișierelor text se va face cu comanda gzip, iar a fișierelor script-uri se va face cu comanda zip, prin păstrarea numelui de bază al fiecărui script și înlocuirea extensiei .sh cu extensia .zip, pentru fiecare script în parte.

(Indicație: folosind comanda basename se poate obține numele fișierului fără extensie. Sau, mai simplu, puteți folosi substituții cu eliminare de sufix.)