



# Proiect UNIX I

ft\_ls

Staff Academy+Plus [contact@academyplus.ro](mailto:contact@academyplus.ro)

*Sumar: Acest proiect are ca scop sa va faca sa retineti comanda "ls".*

# Cuprins

<b>I</b>	<b>Preambul</b>	<b>2</b>
<b>II</b>	<b>Subiect - Parte obligatorie</b>	<b>4</b>
<b>III</b>	<b>Subiect - Partea bonus</b>	<b>5</b>
<b>IV</b>	<b>Instructiuni</b>	<b>6</b>

# Capitolul I

## Preambul

Iata o reteta de Varza murata à la Alsacia:

- Ingrediente (4 persoane)

- 1kg de varza murata
- 350g de backon afumat
- 350g pulpa superioara de porc
- 1 lingura de untura
- 2 catei de usturoi
- 1 frunze de dafin
- 10 boabe de ienupar
- 1 ceapa cu 2 cuisoare
- 25cl de vin Riesling
- 350g de cartofi
- 4 carnati de Strasbourg

- Prepararea

- Clatiti varza murata sub un jet de apa rece. Se scurge, iar apoi se punne jumătate într-o oală.
- amestecati cu varza backon-ul si palette-le, apoi adaugati si restul de varza murata.
- Adaugati untura, usturoiul (Necuratat!), ienuparul, frunzele de dafin si ceapa.
- Stropiti compozitia cu Riesling, si lasati sa fiarba acoperit si la un foc mic, timp de 1 ora.
- Adaugati cartofii si continuati fierberea timp de 50 minute.
- Adaugati carnatii, apoi lasati sa fiarba inca 10 minute.

- Serviti cu o cantitati nerezonabile de bere.

Acest proiect este mai usor daca il veti realiza dupa ce veti manca din Varza murata à la Alsacia

# Capitolul II

## Subiect - Parte obligatorie

- Trebuie sa inregistrati comanda de sistem `ls`.
- Comportamentul acesteia trebuie sa fie identic cu comanda de sistem originala `ls`, cu urmatoarele mici modificari:
  - Dintre optiunile disponibile nu sunteti obligati sa faceti decat `-l`, `-R`, `-a`, `-r` si `-t`.
  - Nu trebuie sa va ocupati de formatarea pe mai multe coloane a rezultatelor de iesire atunci cand optiunea `-l` nu e transmisa.
  - Nu sunteti obligati sa gestionati ACL-urile si attributele extinse.



`man ls`

# Capitolul III

## Subiect - Partea bonus



Bonusurile vor fi evaluate doar daca partea obligatorie este PERFECT facuta. Intelegem prin asta ca este realizata integral, ca gestiune a erorilor se face in mod corespunzator, chiar si in cazurile extreme sau de utilizare gresita.

Iata cateva idei interesante si utile de realizat pentru partea bonus. Evident, si voi puteti adauga functionalitati care vor fi evaluate de catre cei care va vor corecta.

- Gestiunea ACL-urilor si a atributelor extinse
- Gestiunea coloanelor fara a utiliza optiunea -l. (man 4 tty)
- Gestiunea optiunilor -u, -f, -g, -d, ...
- Afisarea in culori (Similara optiunii -G)
- Optimizarea codului (Ce structura de date folositi? Este cea mai eficienta? Chiar si pentru algoritmi de sortare?)

# Capitolul IV

## Instrucțiuni

- Acest proiect va fi corectat doar de persoane. Sunteți deci, liberi, să vă organizați și denumiți fișierele cum doriți, respectând totuși constrângerile de mai jos.
- Fișierul executabil trebuie să se numească `ft_ls`.
- Trebuie să livrați un fișier `Makefile`.
- Acest fișier `Makefile` va trebui să compileze proiectul și trebuie să conțină regulile obisnuite. Programul nu trebuie recompilat decât în caz de necesitate.
- Dacă ești deștept și utilizezi biblioteca `libft` pentru `ft_ls`, va trebui să copiezi fișierul sursă și `Makefile` asociat în directorul denumit `libft` care va trebui să se găsească în rădăcina directorului de lucru. Fișierul `Makefile` trebuie să compileze librăria, apelând `Makefile`-ul corespunzător, apoi să compileze proiectul.
- Proiectul trebuie să fie scris conform standardului de cod (Norme).
- Trebuie să gestionați erorile într-o manieră rezonabilă. În niciun caz programul nu trebuie să se termine neașteptat (Eroare de segmentare etc...). Dacă nu sunteți siguri, gestionați cât mai multe erori ca `ls`.
- Trebuie să livrați, în rădăcina directorului de lucru, un fișier `auteur` conținând login-ul vostru urmat de `'\n'`:

```
$>cat -e auteur
xlogin$
$>
```

- În cadrul părții obligatorii aveți dreptul să folosiți următoarele funcții:
  - `write`
  - `opendir`
  - `readdir`
  - `closedir`

- stat
  - lstat
  - getpwuid
  - getgrgid
  - listxattr
  - getxattr
  - time
  - ctime
  - readlink
  - malloc
  - free
  - perror
  - strerror
  - exit
- Puteti folosi si alte functii in cadrul partii bonus, cu conditia ca acestea sa fie justificate corespunzator in timpul corectarii. De exemplu, folosirea `tcgetattr` este justificata in anumite cazuri, dar folosirea `printf` de lene, niciodata. Fiti destepti.
  - Puteti adresa intrebarile voastre forum, pe jabber, IRC, ...