

# Sisteme de Operare

## Prezentarea cursului

**Cristian Vidraşcu**

<http://www.info.uaic.ro/~vidrascu>

# Cuprins

- Despre obiectivele disciplinei
- Resurse
- Prezentare generală – cursuri și laboratoare
- Examinare
- Feedback

# Objective

## Obiectivele generale urmărite:

1. Dobândirea de cunoștințe despre sistemele de operare, referitoare la tehnicile de proiectare și de implementare a acestora.
2. Deprinderea unor abilități de procesare paralelă și de utilizare a sistemului de operare UNIX/Linux.

# Objective

## Abilități dobândite:

- 1) Însușirea conceptelor de bază referitoare la funcționarea sistemelor de operare
- 2) Înțelegerea arhitecturii unui sistem de operare, cu principalele sale componente
- 3) Înțelegerea algoritmilor care sunt folosiți de un sistem de operare pentru administrarea resurselor
- 4) Utilizarea interfeței text oferită de sistemul de operare UNIX/Linux și a tehnicilor de procesare paralelă
- 5) Proiectarea de aplicații soft care să utilizeze serviciile oferite de un sistem de operare UNIX/Linux.

# Resurse

- Pagina web a cursului Sisteme de Operare:

<http://www.info.uaic.ro/~vidrascu/SO/>

- Cărți:

- A.Tanenbaum, *Organizarea structurată a calculatoarelor*, Ed. Agora, Tg. Mureș, 1999
- F.M.Boian, *Sisteme de operare interactive*, Ed. Libris, Cluj-Napoca, 1994
- A.Silberschatz, P.Galvin, G.Gagne, *Operating Systems Concepts – 9<sup>th</sup> ed.*, Ed. John Wiley & Sons Inc., 2013
- A.Tanenbaum, *Modern Operating Systems – 4<sup>th</sup> ed.*, Ed. Prentice Hall International, 2014

- Documentații și resurse online

- <http://www.linux.org> , ș.a.

# Prezentare generală

## Cursul de Sisteme de Operare

# Prezentare generală /1

- **Concepte de bază**

Istoric. Tipuri de sisteme de operare. Exemple.

- **Structura sistemelor de operare**

Componente. Servicii. Proiectarea și implementarea sistemelor. Nucleu.

- **Gestiunea proceselor**

Concepte. Concurență. Planificare.

- **Coordonarea proceselor**

Secțiuni critice. Sincronizări. Comunicații inter-procese. Interblocaj.

# Prezentare generală /2

- **Administrarea memoriei**

Ierarhii. Alocare. Segmentare. Memorie virtuală. Paginare. Memorie partajată distribuită. Cache. Exemple.

- **Administrarea perifericelor**

Dispozitive de stocare. Sisteme de fișiere. Organizare și operații. Implementare. Protecția fișierelor. Politici.

- **Sisteme de operare distribuite**

Tipuri. Coordonare distribuită. Sisteme de fișiere distribuite. Exemple.



Prezentare generală

# Laboratoarele de Sisteme de Operare

# Prezentare generală /3

## Sistemul de operare UNIX (Linux)

- **Utilizarea sistemului:**
  - Comenzi uzuale
  - Programare shell (bash)
- **Programare concurentă în C:**
  - Fișiere
  - Procese
  - Comunicații inter-procese (canale interne și externe, semnale, redirectări I/E, ș.a.)
  - Gestiunea resurselor (terminale, useri, ș.a.)

## Câteva concepte:

### - Programare secvențială (clasică):

= un program cu un singur flux de instrucțiuni în curs de execuție

vs.

### - Programare **paralelă**, **concurentă** (uneori și **distribuită**):

= un program cu mai multe fluxuri de instrucțiuni în curs de execuție în “același” timp

Cele  $N \geq 2$  fluxuri de instrucțiuni (*threads or sequential processes*) se execută în paralel și, în plus, ele **concurează** pentru utilizarea resurselor oferite de SO-ul acelu sistem de calcul.

Uneori, cele  $N \geq 2$  fluxuri se pot executa într-o manieră **distribuită în spațiu**, i.e. pe sisteme de calcul diferite, interconectate prin rețea.

Cum se pot executa mai multe fluxuri de instrucțiuni în “același” timp?

- prin **parallelism aparent**: tehnica de **multiprogramare** (*multitasking*)

vs.

- prin **parallelism real**: tehnica de **multiprocesare** (*multiprocessing*)

Clasificare după UI (i.e. interfața cu utilizatorul):

**- Program cu UI în mod text** (prescurtat, *TUI*, or *CLI*):

= un program ce interacționează cu utilizatorul printr-o interfață în mod linie de comandă

→ paradigma de programare **imperativă** clasică (*program-driven programming*),  
cu UI realizată prin instrucțiuni uzuale de intrare/ieșire (*input/output*), folosind doar tastatură (+ecran)

**vs.**

**- Program cu UI grafică** (prescurtat, *GUI*):

= un program ce interacționează cu utilizatorul printr-o interfață grafică

→ paradigma de programare **dirijată de evenimente** (*event-driven programming*),  
cu UI realizată folosind diverse biblioteci / *framework*-uri, cu ierarhii formate din sute de clase

Interfața grafică poate fi de două feluri:

- *desktop metaphor*: GUI bazată pe interfețe WIMP (“*windows, icons, menus, pointer*”), folosind tastatură și *mouse*
- *multi-touch metaphor*: GUI bazată pe interfețe post-WIMP, pentru dispozitive cu ecran tactil (*touchscreen*)

# Modul de examinare

## I) Activitatea la laboratoare (Lab):

- 1) exerciții de laborator ExL: punctaj (maxim) 10p
- 2) două teste practice de rezolvat pe calculator în laborator
  - primul test TP1 : (aprox.) în săptămâna a 6-a
  - al doilea test TP2 : (aprox.) în săptămâna a 12-a

Aceste două teste practice **nu** se vor da și în restanțe!

(Notă: le veți putea reface în anul următor cu grupa specială de restanțieri)

- punctaj (maxim) : 25p pentru fiecare test practic

Punctajul total pentru laborator :  $\text{Lab} = \text{ExL} + \text{TP1} + \text{TP2}$

Criteriul de promovare a laboratorului:  $\text{Lab} \geq 20\text{p}$

# Modul de examinare

## II) Examenul scris (TS) :

- Se va susține la finalul semestrului [sau în restanțe] un test scris cu subiecte din partea de teorie
- Condiții de intrare în examen:  
**promovarea laboratorului** și prezența la testul scris
- Punctajul (maxim) TS : 40p
- Criteriul de promovare a examenului:  **$TS \geq 15p$**
- Bonus (pentru motivarea pregătirii practice):  
dacă  **$TP1 \geq 23p$**  și  **$TP2 \geq 22p$**  , atunci NU mai este obligatoriu să susțineți testul scris, acesta fiind echivalat cu  **$TS = (TP1+TP2)*4/5$**

# Modul de examinare

## III) Nota finală:

### - Criteriul de promovare a disciplinei:

să fie promovate laboratorul și examenul scris

### - Punctajul final (PF):

$$PF = TS + Lab$$

Notă: punctajul final se calculează numai pentru cei care îndeplinesc criteriul de promovare a disciplinei

### - Nota finală:

se calculează pe baza PF aplicând “Gauss”, în conformitate cu procentajele adoptate de FII (5% - 10, 10% - 9, 20% - 8, 30% - 7, 25% - 6, 10% - 5)

# Feedback

- Discuții (în clasă/la consultații)
- Pentru orice comentarii, observații, probleme referitoare la S.O. (curs/laboratoare)
  - email la [vidrascu@info.uaic.ro](mailto:vidrascu@info.uaic.ro)
  - contactați echipa S.O. (C.Vidrașcu, I.Leahu, R.Benchea, M.Leonte, A.Simion, D.Moroșanu, A.Munteanu)
- Grupul local LUG

Întrebări ?