

## Curs 4

---

### 1. Comenzi utile pt fisiere

- mkdir => creeaza directoare
  - rmdir => sterge directoare (goale)
  - touch => creaza un fisier gol daca nu exista deja
  - mv => muta directoare sau fisiere
- 

### 2. Wildcards

- ^ => inceputul liniei
  - \$ => sfarsitul liniei
- 

### 3. Cautarea in fisiere

- grep                \$ grep <expresie> <fisiere>
- 

### 4. Procese

Abstractia executiei unui program

- Compuse din instructiuni si date
- Identificat prin Process ID (PID)
- Mai multe instante in rulare ale aceluasi program sunt procese diferite, cu PID-uri diferite
- Executia lor este **secventiala**, nu exista executie paralela a instructiunilor intr-un singur proces
- Format din:
  - Codul program (sectiunea .text)
  - Starea curenta (registrele CPU)
  - Stiva (date temporare)
  - Sectiunile .data
  - Heap
- comanda **ps** afiseaza PID-urilor proceselor aflate in rulare momentan

---

## 5. Comunicare intre procese (IPC)

- Procesele pot fi independente sau cooperante
- Procesele cooperante pot afecta sau pot fi afectate de alte procese, inclusiv prin partajarea datelor

---

## 6. Modele de comunicare

- Memorie partajata
- Schimb de mesaje
  - Operatii: send(), receive()
  - **Directa**: procesele isi folosesc identitatea explicit
    - send(P, message)
    - receive(Q, message)
    - ex: sockets
  - **Indirecta**: mesajele sunt pasate prin casute postale (**porturi**)
    - send(A, message)
    - receive(A, message)

---

## 7. Sincronizare

Schimbul de mesaje poate fi

- **blocant** (sincron)
  - send blocant – transmitatorul e blocat pana cand se primeste mesajul
  - receive blocant – receptorul e blocat pana cand un mesaj e disponibil
- **neblocant** (asincron)
  - send neblocant – transmitatorul trimite mesajul si continua
  - receive neblocant – receptorul primeste:
- Daca send si receive sunt ambele blocante, avem un **\*\*rendezvous\*\***

---

## 8. Semnale

- Notificari asincrone
- Echivalentul software al exceptiilor (HW sau SW)
- Se trimit fie intre procese, fie de catre kernel catre un proces