

# Elaborarea sistemelor operationale

# Scopurile la proiectarea sistemelor operationale

- Pentru crearea unui sistem de operare universal este necesar sa se urmeze urmatoarele puncte:
- Definirea abstractiilor. Abstractiile corespund sincronizarii, semnalelor si modelurilor de memorie, modelarii de intrare/iesire.

Fiecare abstractie poate fi realizata cu ajutorul unor structure de date concrete. Utilizatorii pot crea procese, fisiere si semafoare. Pot opera cu aceste structure cu ajutorul operatiilor simple.

- Realizarea operatiilor primitive
- Asigurarea securității
- Operarea cu unitățile periferice

# Probleme la proiectarea sistemelor de operare(SO)

- In primul rind sistemele de operare au devenit programe destul de mari.
- SO trebuie sa contina parallelism.
- SO trebuie sa ia in considerare ca exista utilizatori, care doresc sa stabileasca modul de lucru a SO.
- Utilizatorii trebuie sa aiba posibilitatea de a accesa fisiera comune, dar si sa stabileasca informatia privata.
- SO trebuie sa aiba perioada de viata destul de lunga.
- SO trebuie sa fie universal.
- SO trebuie sa fie portabil la diferite platforme. Sa poata lucra cu diferite tipuri de dispozitive.
- SO trebuie sa fie compatibil cu versiunile vechi ale SO.

# Elaborarea interface-ului

Principiile de baza:

.Simplitatea

.Complexitatea

.Eficacitatea

Principala problema e ca fiecare functie din SO sa interactioneze perfect intre ele, e esential ca sa se defineasca doua tipuri de utilizatori.

Initial elaborarea partii grafice a sistemului prezinta proiectarea de sus in jos. Trebuie sa se stabileasca ce functii va avea interfaceul si cum va interactiona utilizatorul cu ele.

Elaborarea cererilor sistemului se face de jos in sus, se stabilesc ce functii necesita programistii. Deobicei pentru intretinerea interfaceului grafic nu necesita multe functii speciale.

# Paradigmele

- Paradigmele interfaceului utilizatorului. Majoritatea interfeurilor grafice utilizeaza paradigm WIMP( Windows, Icons, Menus, Pointers) Se creeaza biblioteci si instrumente de acces la procedurile ce permit crearea unui stil inchegat.
- Paradigmele executiei. Pe larg sunt utilizate doua paradigme:
  - Paradigma algoritmica bazata pe idee ca programul se porneste pentru crearea unor functii preventiv cunoscute ori stabilite sub forma parametrilor.
  - Conducerea cu evenimente. Programul efectuiaza o initializare, iar apoi asteapta cind SO va trasmite primul eveniment. Asa schema e buna pentru program ce interactioneaza active cu utilizatorul.
- Paradigma datelor. Defineste cum trebuie sa apara datele sistemice si structurile pentru programist. SO moderne prezinta datele sub forma de obiecte.