



## SO sinteza teorie examen

Sisteme de operare (Academia de Studii Economice din București)

### Ce este un sistem de operare?

- Set de programe ce administrează resursele unui calculator
- O interfață între utilizatori și hardware
- Permite stocarea convenabilă de date; ascunde chestiunile de detaliu
- Permite utilizarea eficientă a sistemului, desfășurarea în paralel a mai multor activități, înlătură irosirea ciclurilor de ceas
- Oferă posibilitatea de protecție a informațiilor
- Oferă fiecărui utilizator o "felie" din totalul resurselor sistemului
- Acționează ca un program de control.

### În linii mari, un SO poate fi văzut ca:

- un mecanism utilizat pentru planificarea job-urilor și a proceselor. Activitatea de planificare poate fi foarte simplă ca în cazul rulării următorului proces dintr-o coadă de așteptare, sau poate fi ceva mai complicată prin folosirea unor reguli mai complexe de alegere a următorului proces ce va fi rulat.
- o modalitate pentru execuția simultană a mai multor UCP și pentru administrarea I/O. Procesarea are loc chiar dacă se pregătește o operație de I/O.

**Multiprogramare** (comutarea dinamică). În timp ce un job așteaptă pentru eliberarea unei resurse, UCP poate căuta să ruleze alt job. Asta înseamnă că mai multe job-uri sunt gata simultan să fie rulate și așteaptă UCP pentru a continua.

**Planificarea UCP** este o parte importantă în studiul SO:

- **managementul memoriei (și al proceselor)**
- **planificarea resurselor**
- **protecția împotriva blocajelor**

### Alte caracteristici:

- **Partajarea timpului** – mediul de multiprogramare este, de asemenea, interactiv
- **Multiprocesarea** - Sisteme ce colaborează și comunică prin intermediul unei memorii partajate. Folosită pentru aplicații științifice. Utilizată pentru îmbunătățirea vitezei
- **Sisteme distribuite** – Sisteme conectate la distanțe mai mari ce comunică prin transfer de mesaje. Avantaje: partajarea resurselor, creșterea vitezei, fiabilitate, comunicație.
- **Sisteme în timp real** – Caracteristica principală este răspunsul rapid. Sunt utilizate pentru controlul aplicațiilor unde un răspuns rapid este esențial.

### Tipuri de sisteme de operare

- **SO de pe smart-carduri**
- **SO embedded** – sunt SO încorporate în dispozitive mobile, televizoare, etc. Exemple: Android, iOS, Windows Phone.
- **SO în timp real (RTOS – Real Time OS)**
- **SO desktop** – familia Windows (7,8,10), Linux (diverse distribuții), Mac OSX (El Capitan).
- **SO server** - Unix/Linux (RHEL, SLES), Windows Server 2012.
- **SO mainframe** – IBM z /OS (z10), Linux, OpenSolaris.

**Întreruperi** - un echipament ce trimite un semnal de "deranj" către UCP pentru a obține un serviciu. Diferit față de acțiunea UCP de **sondare**.

- Depinde de întreruperi pentru a determina pasul următor care trebuie făcut.
- Există întreruperi hardware și software.
- Administratorul de întreruperi alege codul ce trebuie rulat pentru fiecare echipament în parte.

Protecția trebuie să includă:

- - **Partajarea resurselor**
- - **Arhitecturi multiprocesor**
- - **Sisteme cluster**

### Atuuri UNIX:

- Bazat pe standarde
- Puternic, flexibil, scalabil, securizat
- Suport din partea producătorilor de echipamente
- SO matur și stabil
- Bine integrat cu protocoalele de rețea TCP/IP

- Folosit pe scară largă pentru aplicații critice

#### **Componente:**

- Kernel
- Shell
- Sistemul de fișiere
- Comenzi

#### **Kernelul:**

- Centrul SO - asigură capacitățile necesare funcționării calculatorului
- Fișier executabil ce este încărcat atunci când pornește (*bootează*) calculatorul și se numește *unix* (System V) sau *vmunix* (BSD).
- După încărcarea în memorie, kernelul îndeplinește următoarele funcții principale:
  - Administrează echipamente, memoria, procesele
  - Controlează transmisia de informații dintre programele de sistem și hardware-ul sistemului

#### **Kernelul:**

- Administrează funcții legate de:
  - Spațiul de swap – foarte important pentru Unix, rezervat pentru lucrul memoriei virtuale
  - Demoni – programe (procese) ce îndeplinesc o funcție specifică sau monitorizează execuția unor programe sau funcționarea unor echipamente.
  - Sisteme de fișiere – ierarhie de fișiere, directoare și subdirectoare pentru organizarea și administrarea informației pe HD.

**Shell-ul:** Bourne, Korn, C, Bash, TC

#### **Sistemul de fișiere**

**/bin** – comenzi UNIX

**/usr/bin** – comenzi, utilitare de administrare a sistemului, rutine de biblioteci

**/usr/ucb** – comenzi originale ale variantei BSD

**/opt** - aplicații opționale sau provenite de la alți producători

**/etc** – fișiere de administrare a sistemului (vezi fișierul cu parole)

**/dev** – fișiere ce sunt pointeri la nume de echipamente

**(Solaris)/kernel** – conține fișierele de bază ale SO

**/sbin** – executabilele de bază folosite la bootare și la recovery + utilitare de administrare

**/tmp** – fișiere temporare ale utilizatorilor

**/var** – locație pentru joburile imprimantei (print spooling) și mesaje de eroare ale sistemului de mail.

**Comenzi** – în jur de 350-400 de comenzi și utilitare

#### **Kernel-ul:**

- Kernel-ul este cel mai folosit termen pentru descrierea nucleului (centrului) sistemului de operare.
- Acesta reprezintă o mică parte de cod (software) ce este încărcată în memorie atunci când computerul pornește.
- Acest cod conține instrucțiuni ce permit kernel-ului să administreze echipamente hardware, alocarea memoriei, procesele sistem și alte programe.

#### **Interfața utilizator:**

- IU este componenta de interacțiune dintre SO și utilizator.
- IU este asemenea unui interpretor ce interpretează apăsarea unei taste a tastaturii, un click de mouse sau alt input pentru programele respective.
- interfață utilizator grafică (GUI) permite utilizatorului să folosească software-ul folosind obiecte vizuale precum ferestre, meniuri de tip "pull-down", pointeri și simboluri grafice.

#### **Sistemul de fișiere:**

- Într-un sistem de fișiere ierarhic, fișierele se află în containere logice aranjate într-o structură arborescentă.
- Sistemul de fișiere începe cu rădăcina arborelui.
- UNIX și Linux numesc aceste containere "directoare" și "subdirectoare".
- Windows și Macintosh folosesc termenii de "folder" și "subfolder".
- Un tip de sistem de fișiere foarte răspândit este File Allocation Table (FAT).

- Sistemele FAT sunt administrate pe disc de către sistemul de operare.
- Tabela conține o hartă a fișierelor și locul unde sunt stocate acestea pe disc.
- Tabela FAT face referire la clusterelor hard-discului, ce reprezintă unitatea logica de bază a stocării pe disc.
- Un anumit fișier poate fi stocat pe mai multe cluster, dar un cluster poate conține date dintr-un singur fișier.
- Sistemul de operare folosește tabela FAT pentru a găsi toate clusterelor de pe disc unde sunt stocate fișierele.
- Există trei tipuri de sisteme FAT:
  - FAT12
  - FAT16
  - FAT32
- FAT16 și FAT32 reprezintă versiuni îmbunătățite ale sistemului original FAT.

#### Scurta comparatie între SO Windows si Linux

- Windows a fost introdus pe piață pentru a fi “user-friendly”, cu o interfață grafică (GUI), inițial ca SO desktop.
- Rădăcinile Linux-ului încep cu UNIX și cu proiectarea modulară ce a făcut popular Linux-ul printre administratorii de sisteme.
- Interfață în mod text/grafic (ambele)
- Costuri (diferențe)
- Modalitatea de obținere/instalare a SO
- Abilitatea de a rula direct de pe CD
- Disponibilitatea aplicațiilor și modul de obținere a software-ului
- Vulnerabilitatea față de viruși
- Caracteristici de securitate
- Suport multi-utilizator

#### Modelul Client-Server

- Majoritatea aplicațiilor de rețea incluzând aplicațiile Internet -precum World Wide Web (WWW) și e-mail, sunt construite pe baza unei relații client/server.
- Un server oferă servicii de rețea (spre exemplu, e-mail) altor programe denumite clienți.
- Odată pornit, un program server așteaptă să primească cereri din partea programelor client. Dacă este recepționată o cerere corectă, serverul răspunde printr-un mesaj ce conține informația respectivă către client.
- Orice computer poate acționa ca server atât timp cât este conectat la rețea și configurat corespunzător.

#### Caracteristicile unui SO modern:

- Multithreading
- Multiprocesare simetrică
- Sisteme de operare distribuite
- Proiectare orientată obiect

#### Funcția de bază a unui sistem de operare este aceea de a controla hardware-ul calculatorului, mediul de execuție al programelor și interfața utilizator.

Un cont utilizator permite serverului să autentifice utilizatorul și să aloce resursele la care acel utilizator are acces. Sistemele ce oferă această funcționalitate se numesc **sisteme multiutilizator**.

De asemenea, un sistem de operare de rețea este un **sistem multitasking**. Acest lucru semnifică faptul că, intern, sistemul de operare este capabil să execute mai multe sarcini (*tasks*) sau procese în același timp.

Sistemele de operare server realizează acest lucru printr-un cod software de planificare ce este integrat în mediul de execuție. Acest planificator are rolul de a **aloca timpul procesorului, memoria și alte elemente ale sistemului** pentru mai multe sarcini în așa fel încât această alocare să permită **partajarea resurselor sistemului**. Termenul “fir de execuție” (**thread**) descrie un program ce are capacitatea de a se executa independent de altele. Sistemele de operare ce suportă “multithreading” permit programatorilor să proiecteze programe ale căror părți divizate în fire de execuție să fie **executate concurențial**.

**Multitasking-ul cooperativ** reprezintă un mediu în care programele partajează adrese de memorie și pot schimba informații între ele.

**Multitasking preemptiv**- În acest caz, SO controlează alocarea timpului procesorului, iar programele pe 32 de biți rulează în spații separate de memorie. În cazul multitasking-ului preemptiv, un program ce nu respectă regula nu poate monopoliza sistemul, iar dacă se blochează, nu va afecta alte programe.

**SO:**

- Componente
- Apeluri de sistem
- Integrarea componentelor
- Mașina virtuală

**Structura SO:**

- Managementul proceselor-Un **proces este o instanță a unui program în execuție** (un program este pasiv, un proces este activ).
- Managementul memoriei principale
  - o Alocarea/de-alocarea pentru procese, fișiere, I/O.
  - o Administrarea mai multor procese în același timp
  - o Se ține cont de cine utilizează memoria
  - o Deplasarea memoriei proceselor către/de la memoria secundară.
- Managementul fișierelor-Un *fișier reprezintă o colecție de informații definit de creatorul său.*
- Managementul sistemului I/O
- Managementul memoriei secundare
  - o Discuri, benzi magnetice, optice, etc.
  - o Administrarea spațiului liber (paginare/swapping )
  - o Alocarea spațiului pe disc (ce date sunt scrise și unde pe disc)
  - o Planificări de citire/scriere de pe/pe disc
- Conectarea la rețea
- Sistemul de protecție
- Sistemul de interpretare al comenzilor

**Un apel de sistem reprezintă principala modalitate prin care un program utilizator interacționează cu SO.**

**Nucleul NT** se ocupă cu întregul trafic de mesaje ce se desfășoară în cadrul sistemului de operare și rulează peste HAL. Nucleul NT este ocupat în principal cu **manipularea întreruperilor și excepțiilor** pentru comunicația între subsisteme și resursele hardware ale sistemului de operare.

**Administratorul NT (NT Executive)** este compus din nucleul NT la care se adaugă o varietate de subsisteme cunoscute împreună sub numele de **servicii sistem**. Printre aceste servicii se află:

- managerul intrărilor și ieșirilor (managerul I/O);
- managerul apelului de procedură locală;
- managerul de obiecte;
- managerul de procese;
- managerul memoriei virtuale;
- monitorul de securitate.

**Managerul I/O**

- Acesta are în sarcină administrarea tuturor intrărilor și ieșirilor pentru sistemul de operare Windows.

Într-o **“mașină virtuală”** fiecare proces pare să se execute pe propriul procesor și cu propria memorie, echipamente, etc

- **Arhivarea** – Reprezintă combinarea mai multor fișiere într-unul singur, ceea ce facilitează transmiterea acestora
- **Compresia** – Reprezintă reducerea dimensiunilor fișierelor prin eliminarea informației redundante
- **Comprimarea fișierelor** le face să aibă dimensiuni mai mici prin eliminarea informațiilor duplicate, ele putând fi restaurate atunci când este necesar.

Există două feluri de comprimare:

- Lossless
- Lossy

**Un sistem de fișiere este o parte integrantă distinctă a sistemului de operare, ce constă din fișiere, directoare, precum și informațiile necesare pentru accesarea, localizarea (și eventual refacerea) acestora.**

**Memoria cache:**

- Tip de memorie mică, ultra-rapidă, aproape de UCP, ce conține cele mai recent accesate date sau instrucțiuni de cod
  - “Cache hit”
  - “Cache miss”

**Memoria virtuală** - Reprezintă separarea conceptuală a memoriei logice disponibile pentru aplicații față de memoria fizică. În acest mod putem avea o memorie virtuală de dimensiuni mari chiar cu o memorie fizică de dimensiuni reduse.

**Paginarea** reprezintă tehnica prin care memoria fizică este împărțită în blocuri de lungime fixă denumite pagini.

**Algoritmi de înlocuire a paginilor:**

- FIFO
- Inlocuirea optima
- LRU

**Procese:**

Un **program** reprezintă ceva pasiv, static; **procesul** este activ (proces = *instanță a unui program în execuție*).

Exemple de atribute asociate cu un proces: starea, memoria alocată, resurse UCP, progresul înregistrat.

**De ce este nevoie de procese ?**

- Pentru partajarea resurselor logice (fișiere) și fizice (echipamente hardware).
- Pentru îmbunătățirea vitezei de calcul, prin implementarea multiprogramării.
- Asigurarea modularității pentru protecție.

**Principala funcție a unui procesor este aceea de a executa *instrucțiuni mașină* ce se află în memoria principală.**

Putem caracteriza comportamentul unui proces individual prin listarea secvenței de instrucțiuni ce se execută pentru acel proces. O astfel de listă se numește **urma (trace) unui proces**.

**Model de proces cu 5 stări:**

- Nou
- Gata de execuție
- În așteptare
- În execuție
- Suspendat/Terminat

**Există două metode de bază pentru comunicarea între procese:**

- - Memorie partajată – modalitate rapidă/fără transfer de date
- - Transfer de mesaje – modalitate distribuită/o mai bună izolare.

**Comunicația directă:** În acest caz se cunosc transmițătorul și receptorul. Procedura de comunicație este următoarea:

```
send (Process_P, message) ;  
receive (Process_Q , message);  
receive (id, message)
```

**Comunicația indirectă:** Procesele comunică printr-o casuță poștală (mailbox). Procedura arată astfel:

```
open(mailbox_name);  
send (mailbox_name, message);  
receive (mailbox_name, message);
```

**Planificarea proceselor:**

- Modalitatea prin care un proces este atașat procesorului.
- Este centrată în jurul algoritmilor eficienți.
- Proiectarea unui planificator se ocupă cu asigurarea faptului că toți utilizatorii au acces în mod corect la resurse.

**Criterii pentru evaluarea performanțelor:**

- Gradul de utilizare
- Throughput
- Timpul de serviciu

- Timpul de așteptare în coadă
- Timpul de rezidență
- Timpul de rezidență
- Timpul de răspuns
- Timpul de "gândire"

**Scopul principal este acela de a optimiza media acestor timpi.**

**Algoritmi de planificare:**

- FIFO
- Cel mai scurt job primul (SJF):
- Algoritmi preemptivi:
- Planificarea pe bază de priorități:
- Coada circulară (ROUND ROBIN)
- Cozi multi nivel
- Planificarea multi-procesor

**Blocaje:**

- "Excluderea mutuală"
- "Hold and Wait"
- Fără preemție
- Așteptare circulară

**Mecanisme de concurență a proceselor în UNIX:**

- Pipes
- Mesaje
- Memorie partajată
- Semafoare
- Semnale

**Legături soft (legături simbolice)** reprezintă un tip special de fișiere ce fac referire la alte fișiere. Asemănătoare cu shortcut-urile din Windows.

**Legături hard** – sunt legături sub formă de fișiere obișnuite; valoarea numărului de legături este incrementată. Un **model arhitectural** oferă un cadru general de referință pentru problemele legate de comunicațiile în rețea. Un astfel de model este folosit nu doar pentru a explica protocoalele de comunicație, ci și pentru dezvoltarea acestora.

**Modelul OSI:**

De ce un model pe nivele?

- Reduce complexitatea
- Standardizează interfețele
- Facilitează concepția modulară
- Asigură tehnologii interoperabile
- Accelerează evoluția
- Simplifică și ajută procesul de predare/învățare

**Procesul de bootare:**

Reprezintă procesul prin care un computer ajunge din starea "off" în starea de rulare.

Procesul de boot-are are patru etape principale:

1. Etapa \*Firmware\*
2. Etapa \*Bootloader\*
3. Etapa \*Kernel\*
4. Etapa \*Init\*

**Securitate:** garanție/pași care trebuie urmați pentru a proteja un computer și informațiile care sunt stocate pe acesta/termenul nu este absolut în context IT

- Criptarea legăturii poate fi folosită în sensul în care utilizatorii au iluzia că se află într-o rețea privată chiar și atunci când este vorba despre o rețea publică. Această abordare poartă numele de **rețea privată virtuală (virtual private network - VPN)**.

- Un firewall reprezintă un dispozitiv (hardware/software) de control al accesului situat între două rețele sau segmente de rețea. Firewall-ul filtrează tot traficul între rețeaua “internă” protejată și rețeaua/segmentul de rețea “externă” mai puțin protejată.
- Modul de transmisie a datelor în cadrul rețelei VPN se numește **tunel criptat de comunicație** sau **tunel**.

**În procesul de creare și utilizare a partițiilor avem de-a face cu trei etape:**

1. Împărțirea hard-diskului în partiții
2. Formatarea hard-diskului folosind un anumit sistem de fișiere
3. Montare sistemului de fișiere în arborele de directoare

**Sistem de fișiere:** Un sistem de baze de date pentru organizarea fișierelor.

**Tabel:** Baza de date unde sistemul de fișiere stochează informațiile.

**Metadata:** Include atribute ale fișierelor proprietar, locație, *timestamps*, etc.

**Inode:** Număr unic atribuit fiecărui fișier din cadrul sistemului de fișiere.

**Componentele sistemului de fișiere Linux:**

- Superblocul
- Blocul de grup
- Tabela Inode

**Avantajele utilizării partițiilor:**

- Oferă suport pentru mai multe sisteme de operare
- Directoare Home
- Directoare comune cu drepturi de scriere
- Securitate sporită
- Partiții extrem de active

**Partiții:** Diviziunea logică a spațiului disponibil de pe un disc într-una sau mai multe secțiuni ce pot fi accesate în mod independent una față de cealaltă.

**Spațiul de swap:** Este utilizat atunci când dimensiunea memoriei fizice (RAM) este insuficientă pentru rularea programelor. Dacă sistemul are nevoie de mai multă memorie, și memoria RAM este plină, paginile inactive din memorie sunt mutate în spațiul de swap.