

Sisteme de Operare



ș.l.dr.ing. Cristian Amarandei
camarand@cs.tuiasi.ro

Curs #01 - Conținut

- Prezentarea disciplinei
- Noțiuni introductive

Prezentarea disciplinei

Organizare

- Curs
 - 2 ore/săptămână [Total: 28 ore curs]
- Laborator
 - 2 ore/săptămână [Total: 28 ore laborator]

Prezentarea disciplinei

Organizare [2]

□ Evaluare

■ Evaluarea finală

- ponderea în nota finală: **50%**
- Probă scrisă (**notă ≥ 5**)
 - Test grilă
 - 1 sau 2 probleme
 - fără acces la surse bibliografice

■ Evaluarea pe parcurs

- Activitatea la laborator (**notă ≥ 5**)
 - ponderea în nota finală: **20%**
 - Criterii de evaluare: rezolvarea problemelor propuse și calitatea soluțiilor.
- Teste laborator (săptămâna 6 și 14) (**notă ≥ 5**)
 - ponderea în nota finală: **30%**
 - Probă practică
 - Acces la lucrările de laborator

Prezentarea disciplinei

□ Obiectivele cursului

- Introducerea conceptelor de bază ale sistemelor de operare: procese, thread-uri, sincronizare, concurență, fișiere, memorie;
- Analiza sistemelor de operare moderne (Linux, Solaris, Windows)
- Prezentarea unor algoritmi/tehnici de proiectare a aplicațiilor complexe
- Înțelegerea modului în care alegerea diferitelor strategii de proiectare și implementare a unui sistem de operare au implicații asupra utilizării resurselor calculatorului și asupra programelor utilizator

□ Rezultatele învățării

- Înțelegerea mecanismelor fundamentale care stau la baza funcționării unui sistem de calcul (hardware -> sistem de operare -> aplicații)
- Cunoștințe despre algoritmi/tehnici de proiectare a aplicațiilor complexe
- Dezvoltarea unor aplicații care utilizează apeluri sistem
- Utilizarea/administrarea mai eficientă a sistemelor de calcul

De ce studiem Sistemele de Operare?

- ▣ **Pentru a învăța cum funcționează sistemele de calcul**
 - ▣ Arhitectura hardware/software
- ▣ **Administrarea/utilizarea eficientă a unui sistem de calcul**
 - ▣ CPU, memorie, sisteme de fișiere, etc.
- ▣ **Asigurarea performanței sistemului/aplicațiilor**
 - ▣ Este necesară modificarea parametrilor SO – nu se poate face fără a înțelege funcționarea acestuia.
 - ▣ Înțelegerea serviciilor oferite de SO influențează modul de proiectare a aplicațiilor complexe
 - ▣ Erori de funcționare ale SO -> sistem de calcul inutilizabil.
 - ▣ Trebuie să aibă mai puține erori de funcționare decât aplicațiile
- ▣ **Depanarea aplicațiilor**
- ▣ **System/low-level/kernel programming**
- ▣ **Anumite aspecte/tehnici pot fi aplicate în alte domenii**
 - ▣ concurență, gestiunea resurselor, tratarea erorilor, gestiunea de structuri complexe (nivelul de abstractizare furnizat de SO)
- ▣ **Securitatea SO -> baza securității sistemului de calcul**
- ▣ **Deoarece sunt peste tot ! (PC, servere, smartphones)**

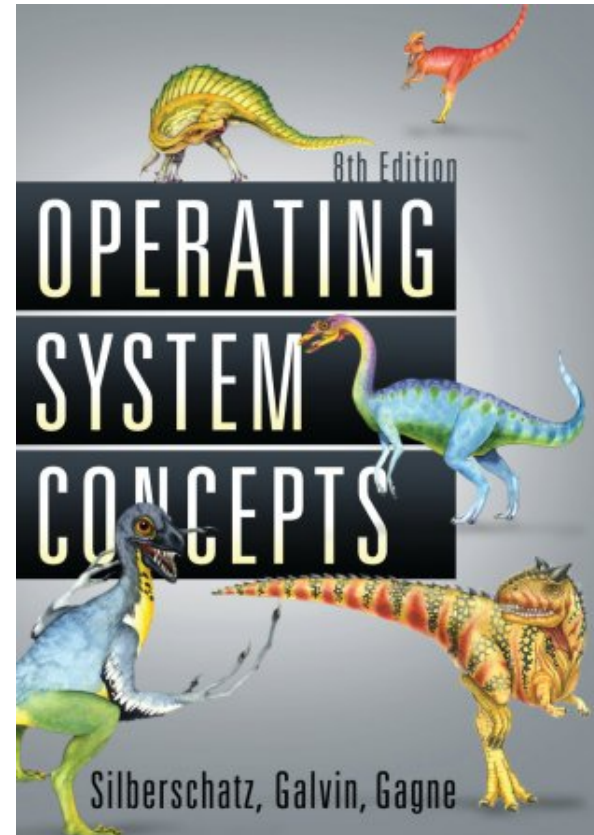
- ▣ **Why the operating system matters even more in 2017**
 - <https://opensource.com/16/12/yearbook-why-operating-system-matters>

Cuprins

- ❑ Noțiuni introductive
- ❑ Structura unui sistem de operare
- ❑ Gestiunea proceselor
- ❑ Gestiunea memoriei
- ❑ Gestiunea sistemului de fișiere
- ❑ Sistemul de I/O

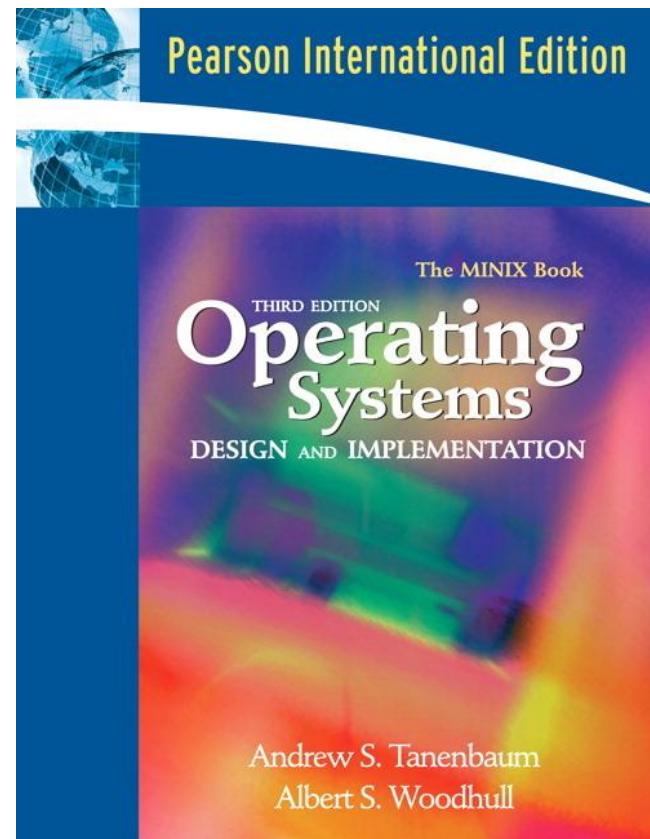
Bibliografie

- Silberschatz A.,
Galvin P. -
**Operating System
Concepts**, 9th
Edition , John Wiley
& Sons, 2012
- [http://www.os-
book.com/](http://www.os-book.com/)



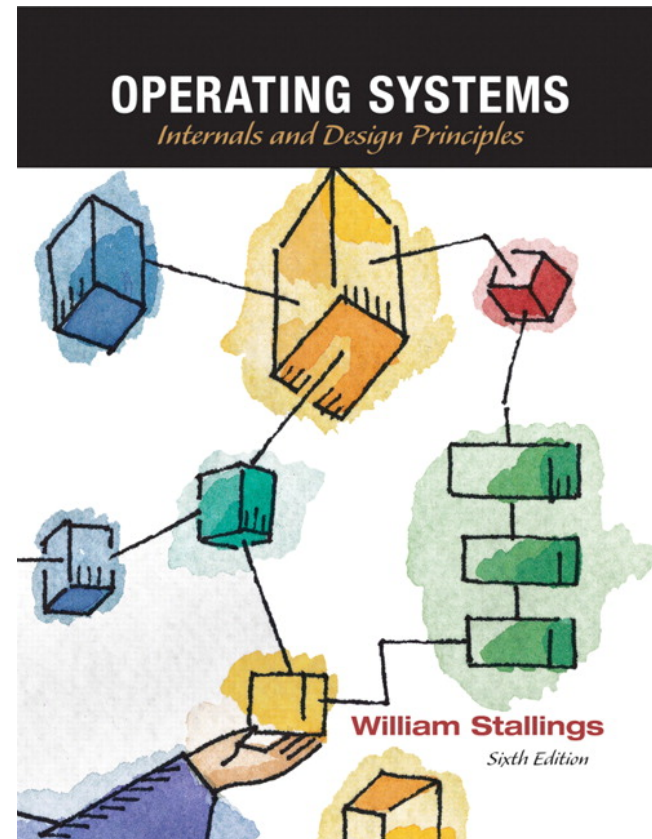
Bibliografie

- Andrew S. Tanenbaum - **Operating Systems: Design and Implementation**, 3rd Edition, Prentice Hall, 2009
 - https://github.com/citruslee/Studijne-materialy-FIIT/raw/master/3._Semester/OS/Materialy/Operating_Systems_Design_%26_Implementation_3rd_Edition.pdf
 - <http://www.minix3.org/documentation/index.html>



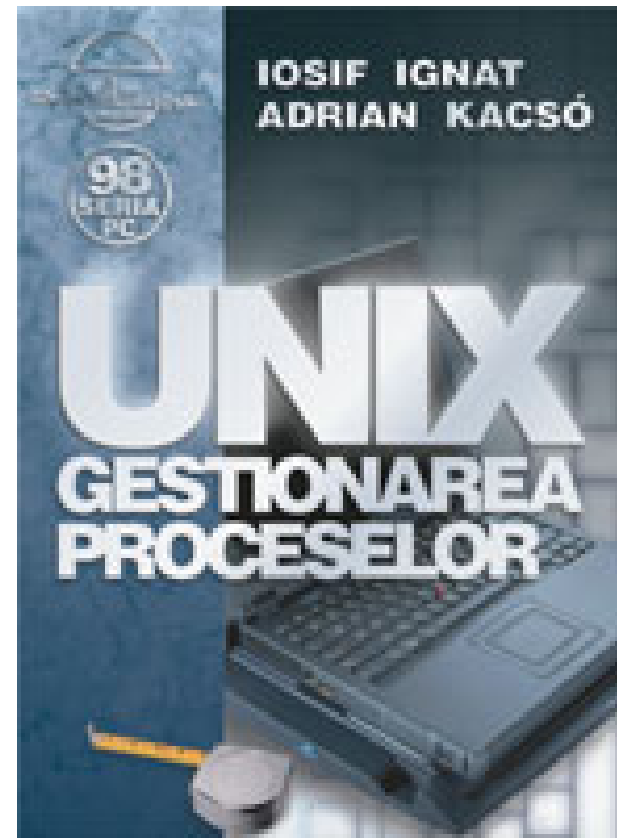
Bibliografie

- William Stallings - **Operating Systems: Internals and Design Principles**, 6th Edition, Prentice Hall, 2008
- <http://williamstallings.com/OS/OS6e.html>



Bibliografie

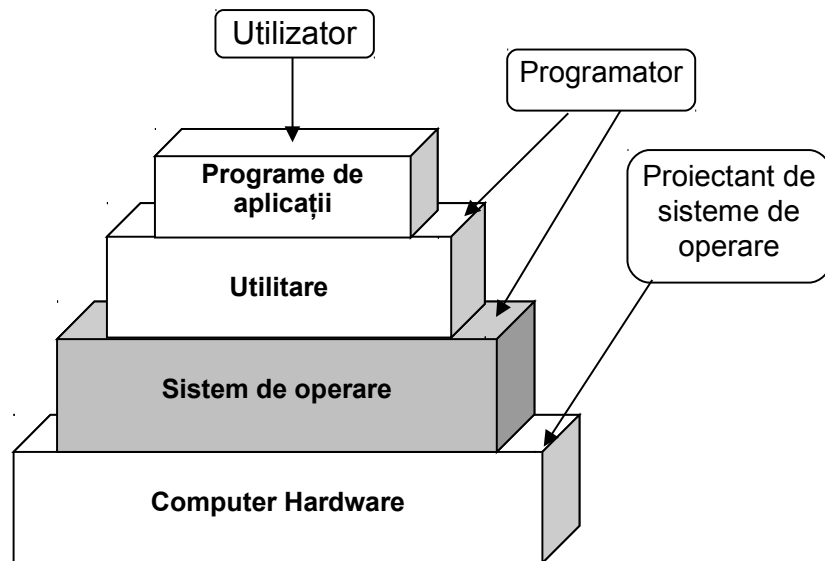
- David Solomon, Mark Russinovich - **Microsoft Windows Internals**, 4th/5th/6th Ed., 2004/2009/2012
- Boian F.M. - **Sisteme de operare interactive**, Ed. Libris, 1994
- T. Ionescu, D. Saru, J. Floroiu - **Sisteme de operare. Principii și funcționare**, Ed. Tehnică, 1997
- Ignat I. - **Unix - Gestionarea Proceselor**, Ed. Albastră, 2006



Definiția unui sistem de operare

- Un sistem de operare reprezintă un set de programe care asigură gestionarea resurselor unui sistem de calcul implementând algoritmi destinați să maximizeze performanțele și realizează o interfață între utilizator și sistemul de calcul, extinzând dar și simplificând setul de operații disponibile.

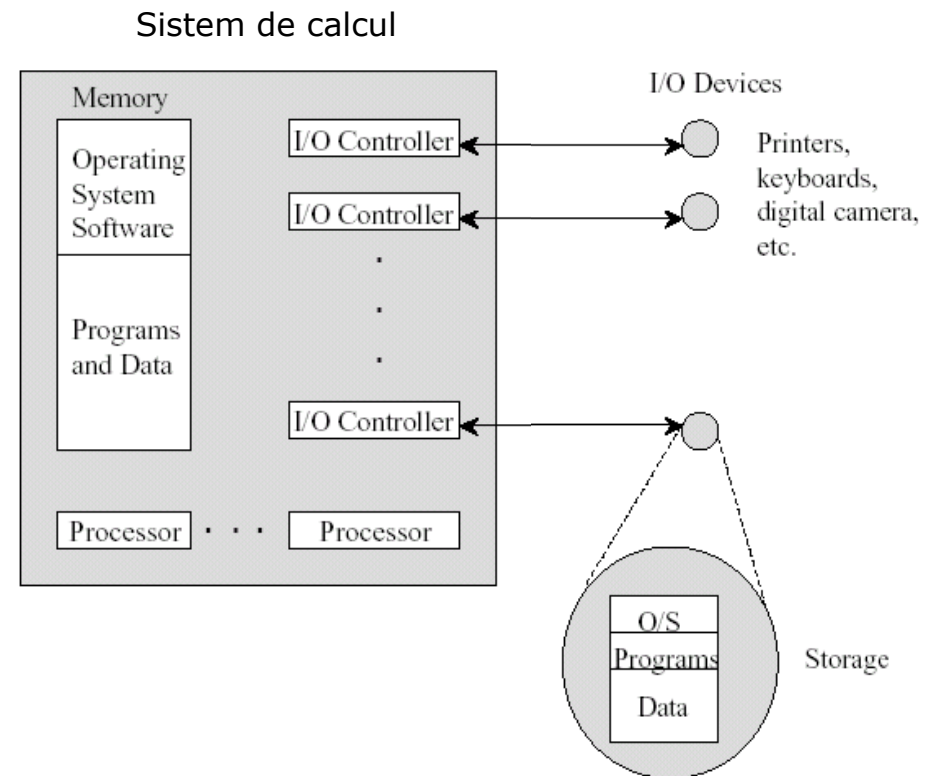
Structura unui sistem de calcul



- **Hardware :**
 - CPU, memorie, dispozitive I/O
- **Sistem de operare:**
 - Controlează și coordonează utilizarea resurselor hardware
- **Programe utilitare**
 - folosite pentru administrare și operarea sistemului de calcul
- **Aplicații:**
 - procesoare de text, compilatoare, web browser, baze de date, jocuri
- **Utilizatori**

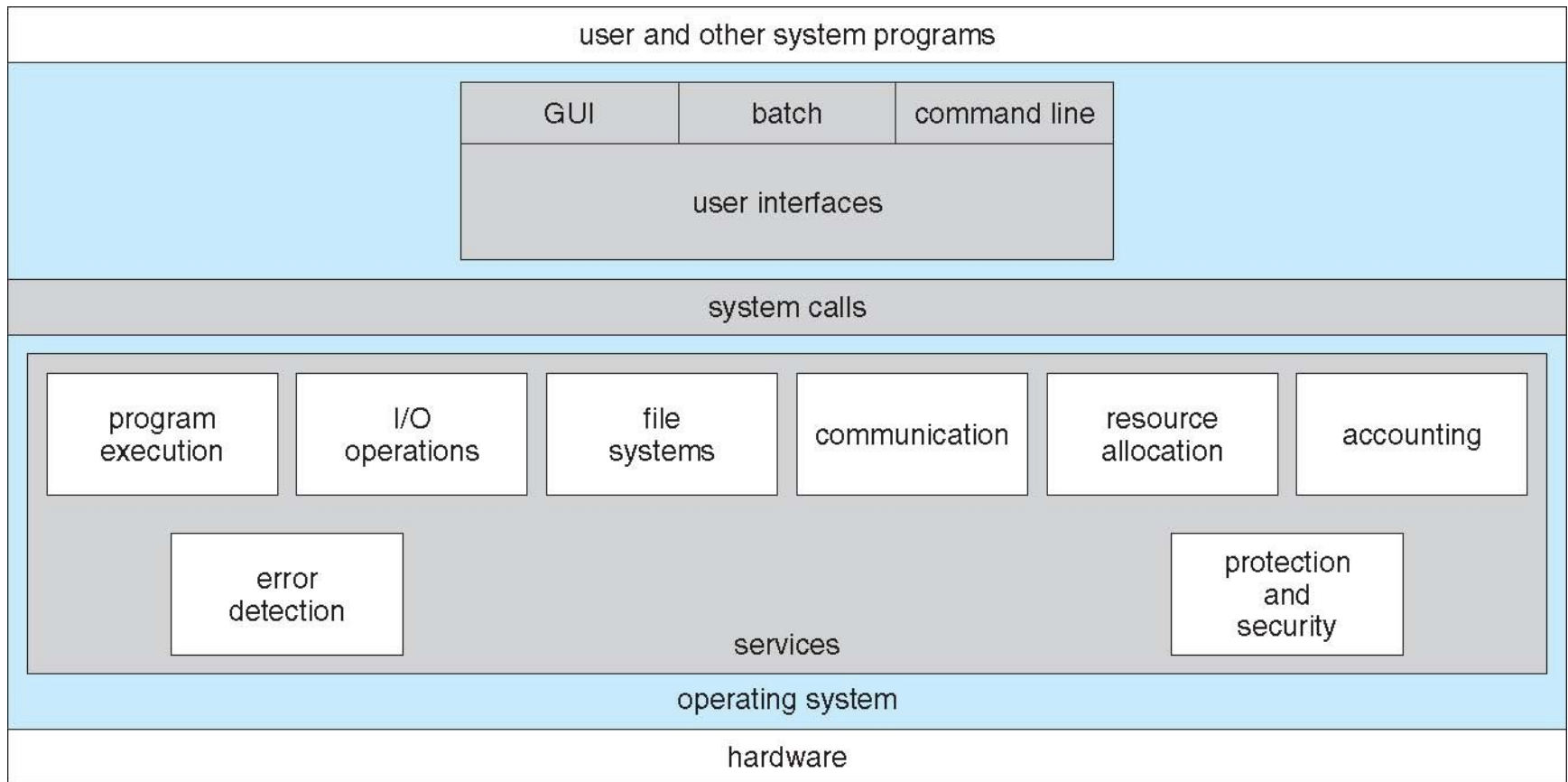
Funcțiile unui sistem de operare

- ❑ Interfața cu utilizatorul
- ❑ Gestiunea fișierelor
- ❑ Gestiunea perifericelor
- ❑ Gestiunea memoriei
- ❑ Gestiunea proceselor
- ❑ Tratarea erorilor
- ❑ Gestiunea sistemului



Sistemul de operare ca manager de resurse

Structura unui sistem de operare



Interfața cu utilizatorul

□ Comenzile

- determină într-o mare măsură acceptarea de către utilizatori a unui sistem de operare
- introduse de utilizatori
- sunt prelucrate de interpretorul de comenzi

□ Apelurile de sistem

- definesc arhitectura unui sistem de operare
- asemănătoare apelurilor de proceduri
- sunt folosite de programatorii de sistem și programatorii de aplicații

Gestiunea fișierelor

- Fișierele reprezintă forma în care sunt păstrate informațiile
- gestiunea fișierelor - modul în cadrul sistemului de operare:
 - operațiile de creare și ștergere a fișierelor
 - controlul accesului la fișiere
 - citirea și scrierea de informații din și în fișiere
 - securitatea informațiilor
 - organizarea colecției de fișiere

Gestiunea perifericelor

- ❑ pregătirea operației
- ❑ lansarea cererilor de transfer de informație
- ❑ controlul transferului propriu-zis
- ❑ tratarea erorilor

Gestiunea memoriei

- o porțiune este rezervată pentru sistemul de operare
- restul memoriei este disponibilă pentru programele utilizator
- Trebuie să rezolve probleme legate de:
 - protecția memoriei folosite de programe
 - împărțirea memoriei disponibile între programele solicitante
 - planificarea schimburilor cu memoria

Gestiunea proceselor

- Trebuie să rezolve probleme legate de:
 - Stările unui proces
 - Contextul unui proces
 - Sincronizarea proceselor
 - Tratarea blocajelor
 - Planificarea proceselor

Tratarea erorilor

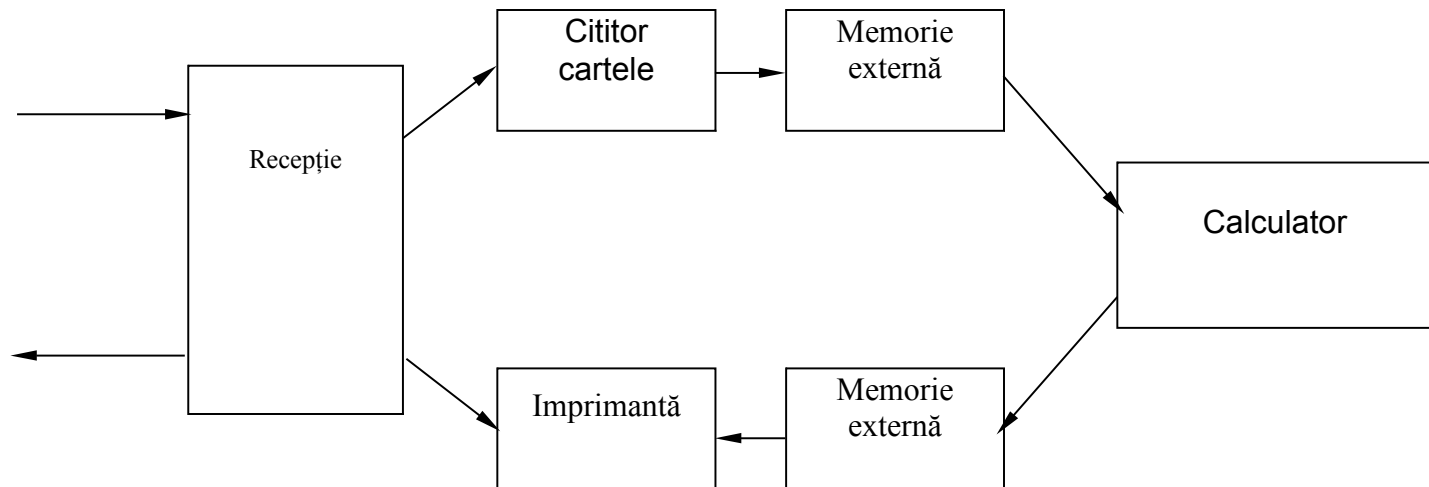
- ❑ sistemul de operare trebuie să reacționeze la o diversitate de erori ce pot apărea atât din cauze hardware cât și software
- ❑ erorile trebuie să fie transparente utilizatorului
- ❑ SO trebuie să aibă mai puține erori de funcționare decât aplicațiile

Evoluția sistemelor de operare

■ **a) Sistemele de tip Batch (SB)** sau cu prelucrare pe loturi – deserveste job-uri dintr-o coada de job-uri.

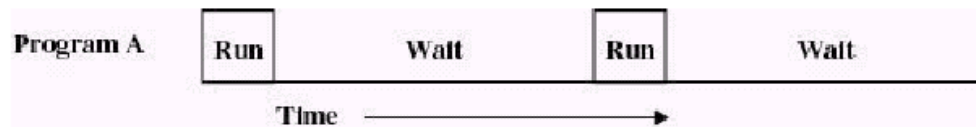
■ Dezavantaje:

- lipsa interacțiunii cu utilizatorul în timpul rulării programului
- timpul de rulare foarte mare

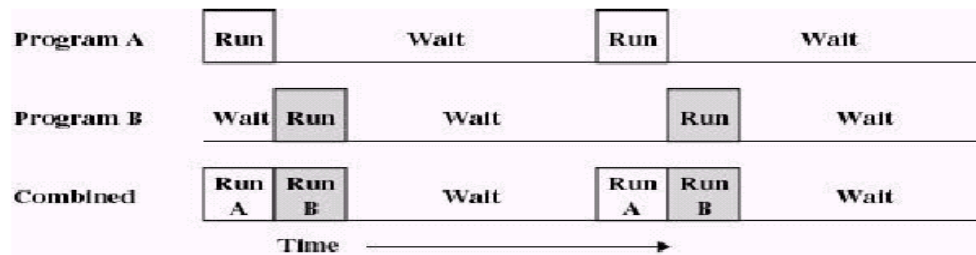


Evoluția sistemelor de operare (2)

- sistemele de tip Batch
 - cu monoprogramare

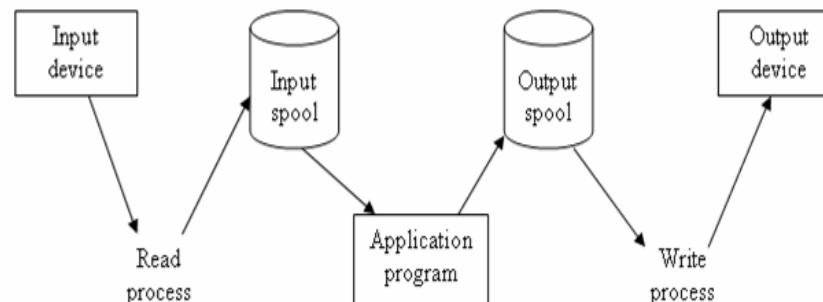


- cu multiprogramare

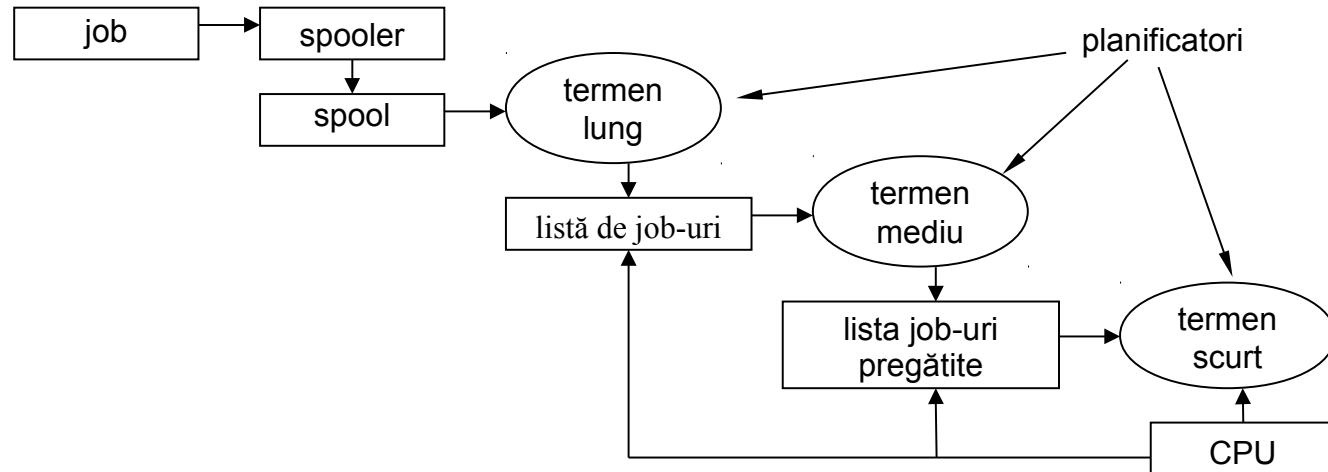


Evoluția sistemelor de operare (3)

- ❑ Sistemele cu multiprogramare au introdus tehnica job spooling
 - SPOOL = Simultaneous Peripheral Operations On-Line.
 - ❑ disponibilitatea unei memorii externe de capacitate mare și cu acces direct
 - ❑ asigurarea unei încărcări la întreaga capacitate a dispozitivelor periferice de intrare/ieșire



Evoluția sistemelor de operare (4)



Evoluția sistemelor de operare (5)

□ **b) Sisteme cu divizarea (partajarea) timpului (time sharing) – SPT**

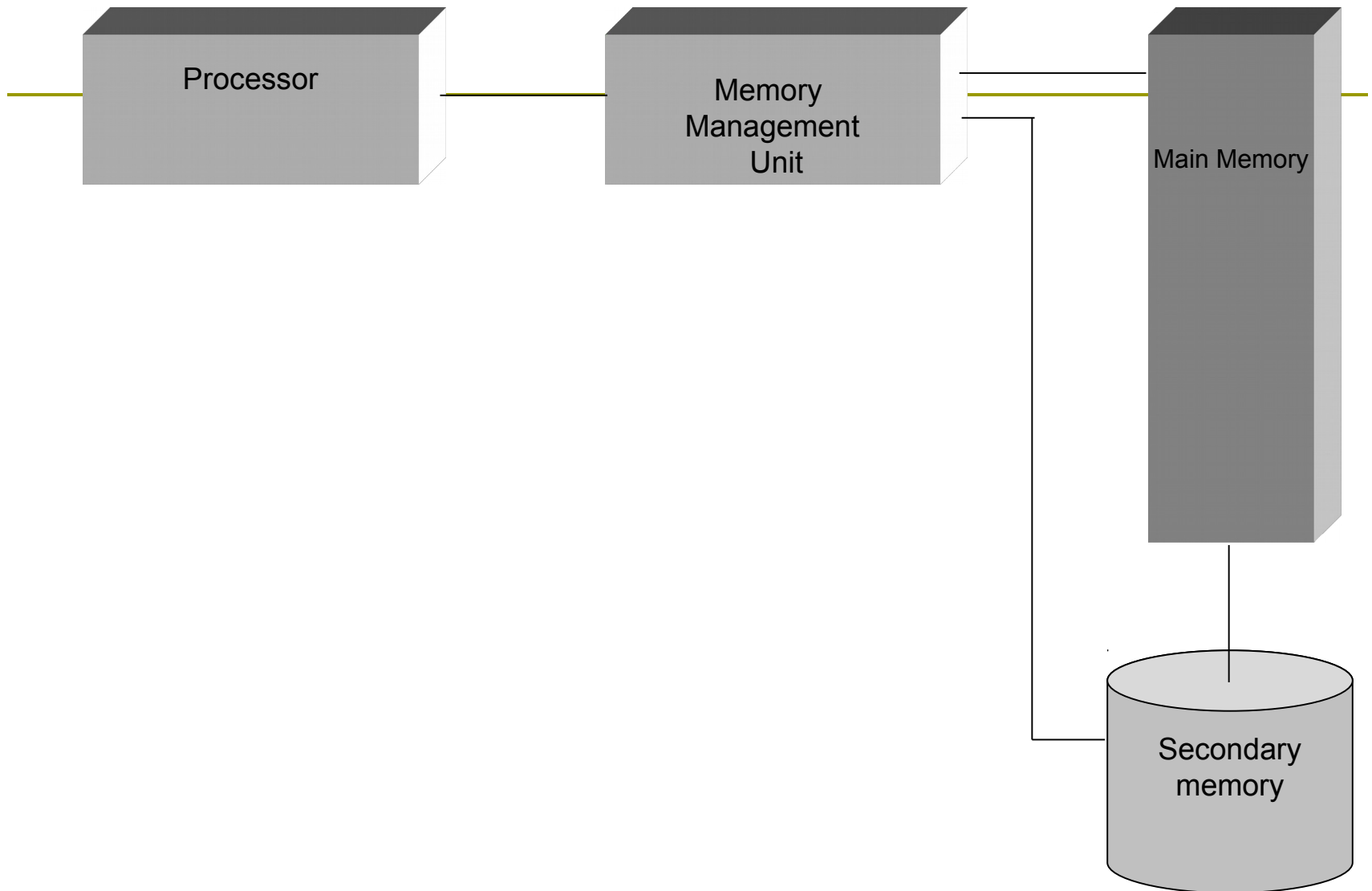
- posibilitatea ca mai mulți utilizatori să partajeze o mașină prin întrețeserea job-urilor
- nu permiteau interacțiuni în timp real între utilizator și sistemul de calcul
- asigură câte un terminal pentru fiecare utilizator al sistemului și din acest motiv se mai numesc sisteme multitasking și multiuser
- se introduce un mecanism de protecție și securitatea accesului la resurse.
- Exemple:
 - CTSS (Compatible Time-Sharing System) dezvoltat la MIT pe un sistem IBM 7094
 - TSS/360 dezvoltat de IBM pentru IBM System 630/67
 - MULTICS , dezvoltat de MIT, AT&T Bell Laboratories și General Electric pe un calculator GE 635 modificat

Evoluția sistemelor de operare (6)

□ **c) Sisteme cu memorie virtuală**

- Prin tehnica de memorie virtuală un disc se poate folosi ca suport automat (fără intervenția programatorului) pentru extinderea memoriei interne, permițând programelor să aibă la dispoziție un spațiu de memorie virtual mai mare decât cel al memoriei interne.
- Exemple:
 - OS/MVS (Multiple Virtual Spaces) ;
 - VM/370 ("Virtual Machine for the 370") - realiza virtualizarea unui întreg sistem nu numai a memoriei.

Sisteme cu memorie virtuală



Evoluția sistemelor de operare (7)

□ d) **Sisteme de operare pentru minicalculatoare**

■ PDP -11(anii `70-80):

- RT-11 - gândit să deservească un singur utilizator, dar prevăzut cu o formă limitată de multiprogramare (foreground / background)
- RSX-11 - un limbaj de comandă puternic, un sistem de fișiere dezvoltat, gestionarea memoriei interne prin partiții de dimensiune variabilă și multiprogramare.

■ PDP-7 - Unix :

- - dezvoltat la Bell Laboratories de Thompson și Ritchie, reimplementat și extins pe PDP-11
- - Inspirat din MULTICS, permite accesul simultan al mai multor utilizatori, fiecare putând lansa procese concurente

Evoluția sistemelor de operare (8)

□ e) **Sisteme de operare pentru calculatoare personale**

- 1971 - microprocesor Intel4004, apoi apar 8008, 8080, urmate de Z80 și Motorola M6800.
- CP/M:
 - primul sistem de operare pe 8 biți
 - produs de Digital Research
 - sistem monoutilizator cu interfață de comandă simplă, sistem de fișiere mononivel, interfață I/E bine dezvoltată
- microprocesoarele pe 16 biți (Intel 8086, Motorola 68000)
 - MS-DOS, Windows 3.1, 3.11
 - monoutilizator, monoproses, sistem ierarhic de fișiere, interfață cu utilizatorul bine dezvoltată.

□ f) **Sisteme de operare pentru sisteme de calcul distribuite**

- au apărut odată cu dezvoltarea rețelelor de calculatoare
 - se urmărește integrarea cât mai puternică a resurselor din rețea astfel încât localizarea resurselor să fie transparentă pentru utilizatori și aplicații (CHORUS, MACH, AMOEBA).

Clasificarea sistemelor de operare

- Sistemul de operare poate fi privit ca o mașină abstractă ce are mai multe niveluri:
 - A0 – nivelul hardware
 - A1 – nivelul microprogram (funcții BIOS)
 - A2 – nivelul funcțiilor realizate în limbaj de asamblare ce gestionează regiștrii, controller-ele, întreruperile, memoria.
 - An-1 – API interfața de apeluri sistem a S.O.

Clasificarea sistemelor de operare (2)

- După calculatoarele pe care rulează:
 - stații de lucru, în general mono-utilizator.
 - Ex: MS-DOS, WINDOWS, sistem UNIX cu facilități grafice
- pentru o rețea de calculatoare poate fi gestionată de un S.O. de tip:
 - file-server (Netware, Unix):
 - S.O. este pe un singur calculator, stațiile având încărcate în memorie o versiune de DOS – la faza de login singura resursă comună – harddisk-ul de pe server
 - peer – to – peer :
 - Windows 3.11 (nodurile sunt echivalente) – același S.O. este încărcat de fiecare calculator ;

Clasificarea sistemelor de operare (3)

□ După accesul la memorie:

- **UMA (Uniform Memory Access)**: adică timpii de acces la memorie sunt egali pentru toate procesele din sistem (memorie globală)
- **NUMA (Non Uniform Memory Access)**: timpi de acces diferiți
- **NORMA (No Remote Memory Access)**: sistemul este cuplat într-o rețea fără memorie globală.

□ Din punctul de vedere al transmisiei de mesaje:

- **Sisteme de operare cu transmitere de mesaje** – permite realizarea unui S.O. distribuit (comunicare între procese atât de pe aceeași mașină cât și de pe mașini diferite).
- **Sisteme de operare fără transmitere de mesaje** –apelul procedurilor pe care le executa procesele utilizator se transmit prin apeluri de proceduri locale.

Clasificarea sistemelor de operare (4)

- Sistemele de operare de timp real
 - Trebuie să opereze în cadrul unor constrângeri de timp
- Clasificare:
 - După comportarea în timp real al task-urilor:
 - task-uri sporadice: task-ul este asincron și nepredictibil (ex: intreruperile)
 - task-uri aperiodice: are un anumit ciclu dar cu perioade de timp variabile (nu respectă o distribuție)
 - task-uri periodice: au o ciclicitate regulată (reîmprospătarea imaginii)
 - După importanța deadline-ului
 - hard - real time: acțiunea unui task ce depășește deadline-ul este 0 sau negativă
 - soft - real time: dacă acțiunea unui task depășește deadline-ul are o valoare indiferentă
- Exemple: QNX, CHORUS

Clasificarea sistemelor de operare (5)

□ Sistemele de operare de timp real

