Arhitectura calculatoarelor şi sisteme de operare

Prof. dr. Henri Luchian
Lect. dr. Vlad Rădulescu

Evaluare

- examinare
  - două teste scrise din materia de curs
    - câte unul pentru fiecare jumătate de semestru
  - un test practic la laborator
    - limbaj de asamblare
- condiţia pentru susţinerea testelor scrise
  - prezenţa la laborator
    - cel mult 2 absenţe permise în fiecare jumătate de semestru

Cuprins - prima jumătate

I. Introducere

I.1. Evoluţie

Cum definim noţiunea de calcul?

- ce operaţii se pot realiza?
- evoluţie în timp
  - abacul: adunări
  - roţi dinţate (Leibniz, Pascal): adunări, înmulţiri
  - Babbage: instrucţiuni încărcate din exterior, calcul ramificat
  - von Neumann: program memorat; execuţie în secvenţă de instrucţiuni; ierarhii de memorii
  - calcul paralel, cuantic etc.

Maşini de calcul universale

- o maşină de calcul universală se poate comporta ca oricare maşină de calcul particulară
  - deci poate rezolva orice problemă pe care o poate rezolva o maşină de calcul particulară
- exemplu - calculatorul
  - în funcţie de programul executat, rezolvă probleme de: calcul matricial, grafică, tehnoredactare etc.

Scurtă istorie (1)

- scrierea poziţională
  - indieni, arabi
- algebra booleană
  - George Boole, 1854
- teorema de incompletitudine
  - Kurt Gödel, 1935
- legătura între algebra booleană şi circuite
  - Claude Shannon, 1938

Scurtă istorie (2)

- calculatorul neumannian
  - John von Neumann, 1946
- tranzistorul
  - Shockley, Brittain, Bardeen, 1947
- circuitele integrate

I.2. Legi empirice

Legi empirice

- în orice domeniu al ştiinţei, legile sunt determinate într-un fel sau altul de experiment sau de observaţii în lumea concretă
- repetabilitatea duce la ideea de legi empirice: adevăruri valabile de cele mai multe ori, conform observaţiilor

Legi empirice în informatică

- legea "90:10" (Donald Knuth)
  - 90% din timpul de execuţie al unui program este utilizat pentru 10% din instrucţiuni
- legea lui Amdahl
  - eficienţa maximă în îmbunătăţirea unui sistem (concret sau abstract) se atinge dacă se optimizează subsistemul cel mai folosit
- legile localizării - spaţială, temporală

Legea lui Amdahl (1)

- considerăm un sistem (hardware, software) şi o anumită componentă a sa
- componenta respectivă lucrează un procentaj f_a din timpul de lucru al sistemului
- şi este îmbunătăţită, astfel încât lucrează de a ori mai rapid decât înainte
- de câte ori mai rapid devine sistemul?

Legea lui Amdahl (2)

$$A(a, f_a) = \frac{1}{(1 - f_a) + \frac{f_a}{a}}$$

- creştere de viteză generală cât mai mare
  - îmbunătăţirea pronunţată a componentei (a)
  - îmbunătăţirea componentelor cu o pondere (f_a) cât mai mare
    - deci mai des folosite