

**FIȘA DISCIPLINEI****1. Date despre program**

| | |
|---------------------------------------|--|
| 1.1 Instituția de învățământ superior | Universitatea “Alexandru Ioan Cuza” din Iași |
| 1.2 Facultatea | Facultatea de Informatică |
| 1.3 Departamentul | Informatică |
| 1.4 Domeniul de studii | Informatică |
| 1.5 Ciclul de studii | Licență |
| 1.6 Programul de studii / Calificarea | Informatică |

2. Date despre disciplină

| | | | | | | | |
|--|---|--------------|---|---------------------|---|-------------------------|----|
| 2.1 Denumirea disciplinei | Arhitectura calculatoarelor și sisteme de operare | | | | | | |
| 2.2 Titularul activităților de curs | Lect. dr. Vlad Rădulescu | | | | | | |
| 2.3 Titularul activităților de seminar | Lect. dr. Vlad Rădulescu | | | | | | |
| 2.4 An de studiu | I | 2.5 Semestru | 1 | 2.6 Tip de evaluare | E | 2.7 Regimul disciplinei | OB |

* OB – Obligatoriu / OP – Opțional

3. Timpul total estimat (ore pe semestru și activități didactice)

| | | | | | |
|--|----|--------------------|----|-----------------------|-----|
| 3.1 Număr de ore pe săptămână | 4 | din care: 3.2 curs | 2 | 3.3 seminar/laborator | 2 |
| 3.4 Total ore din planul de învățământ | 56 | din care: 3.5 curs | 28 | 3.6 seminar/laborator | 28 |
| Distribuția fondului de timp | | | | | ore |
| Studiu după manual, suport de curs, bibliografie și altele | | | | | 14 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren | | | | | 14 |
| Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri | | | | | 28 |
| Tutoriat | | | | | |
| Examinări | | | | | 3 |
| Alte activități | | | | | |
| 3.7 Total ore studiu individual | | | | | 56 |
| 3.8 Total ore pe semestru | | | | | 115 |
| 3.9 Număr de credite | | | | | 5 |

4. Precondiții (dacă este cazul)

| | |
|-------------------|---|
| 4.1 De curriculum | - |
| 4.2 De competențe | - |

5. Condiții (dacă este cazul)

| | |
|--|---|
| 5.1 De desfășurare a cursului | - |
| 5.2 De desfășurare a seminarului/ laboratorului | - |



6. Competențe specifice acumulate

| | |
|--------------------------------|--|
| Competențe profesionale | C1. Descrierea de concepte, teorii și modele folosite în domeniul de aplicare. C2. Utilizarea modelelor și instrumentelor informatice și matematice pentru rezolvarea problemelor specifice domeniului de aplicare. C3. Elaborarea codurilor sursă adecvate și testarea unitară unor componente într-un limbaj de programare cunoscut, pe baza unor specificații de proiectare date. C4. Identificarea de metodologii adecvate de dezvoltare a sistemelor software. |
| Competențe transversale | CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională. CT2. Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională. |

7. Obiectivele disciplinei (din grila competențelor specifice acumulate)

| | |
|----------------------------------|--|
| 7.1 Obiectivul general | Înțelegerea principiilor și tehnologiilor hardware și software care stau la baza dezvoltării sistemelor de calcul. |
| 7.2 Obiectivele specifice | La sfârșitul cursului, studenții trebuie să cunoască: - elementele de bază ale arhitecturii și organizării sistemelor de calcul - fundamentele reprezentării interne ale datelor în calculator - funcțiile principale ale sistemelor de operare |

8. Conținut

| 8.1 | Curs | Metode de predare | Observații (ore și referințe bibliografice) |
|------------|---|---|---|
| 1 | Introducere. Legi empirice. Circuite combinaționale și funcții booleene | expunere, dezbateri, studii de caz, exerciții | - |
| 2 | Minimizare. Circuite combinaționale | expunere, dezbateri, studii de caz, exerciții | - |
| 3 | Circuite secvențiale | expunere, dezbateri, studii de caz, exerciții | - |
| 4 | Reprezentări interne. Reprezentări în virgulă fixă | expunere, dezbateri, studii de caz, exerciții | - |
| 5 | Reprezentări în virgulă fixă. Depășiri. Reprezentări în virgulă mobilă | expunere, dezbateri, studii de caz, exerciții | - |
| 6 | Reprezentări în virgulă mobilă. Arhitectura și organizarea calculatorului | expunere, dezbateri, studii de caz, exerciții | - |



| | | | |
|----|--|---|---|
| 7 | Memoria. Ierarhia de memorii. Memoria cache | expunere, dezbateri, studii de caz, exerciții | - |
| 8 | Recapitulare | expunere, dezbateri, studii de caz, exerciții | - |
| 9 | Unitatea centrală de procesare. Pipeline | expunere, dezbateri, studii de caz, exerciții | - |
| 10 | Îmbunătățirea performanței procesoarelor | expunere, dezbateri, studii de caz, exerciții | - |
| 11 | Arhitecturi paralele de calcul. Dispozitive periferice | expunere, dezbateri, studii de caz, exerciții | - |
| 12 | Sistemul de întreruperi. Sistemul de operare. Gestiunea proceselor | expunere, dezbateri, studii de caz, exerciții | - |
| 13 | Gestiunea memoriei | expunere, dezbateri, studii de caz, exerciții | - |
| 14 | Memoria virtuală. Crearea și execuția programelor | expunere, dezbateri, studii de caz, exerciții | - |

Bibliografie

J. L. Hennessy, D. A. Patterson, *Computer Architecture - A Quantitative Approach*, Morgan Kaufmann Publishers, 1990.

D. A. Patterson, J. L. Hennessy, *Organizarea și proiectarea calculatoarelor. Interfața hardware/software*, Ed. All, 2002.

R. W. Hockney, C. R. Jesshope, *Calculatoare paralele*, Ed. Tehnică, 1991.

A. Tanenbaum, *Organizarea structurată a calculatoarelor*, Ed. Agora, 1999.

A. Tanenbaum, *Modern Operating Systems*, Prentice Hall, 2001.

L. Vințan, *Fundamente ale arhitecturii microprocesoarelor*, ed. Matrix Rom, 2016.

| 8.2 | Seminar / Laborator | Metode de predare | Observații (ore și referințe bibliografice) |
|-----|--|--------------------------------|--|
| 1 | Transformări între bazele de numerație. Legea lui Amdahl | expunere, dezbateri, exerciții | - |
| 2 | Minimizarea funcțiilor booleene | expunere, dezbateri, exerciții | - |
| 3 | Circuite combinaționale | expunere, dezbateri, exerciții | - |
| 4 | Circuite bistabile | expunere, dezbateri, exerciții | - |
| 5 | Reprezentări în virgulă fixă | expunere, dezbateri, exerciții | - |
| 6 | Reprezentări în virgulă mobilă | expunere, dezbateri, exerciții | - |
| 7 | Introducere în limbajul de asamblare Intel x86 | expunere, dezbateri, exerciții | - |
| 8 | Recapitulare | expunere, dezbateri, exerciții | - |
| 9 | Instrucțiuni aritmetice și logice | expunere, dezbateri, exerciții | - |
| 10 | Salturi | expunere, dezbateri, exerciții | - |



| | | | |
|----|-------------------------------------|--------------------------------|---|
| 11 | Lucrul cu stiva. Apeluri de funcții | expunere, dezbatere, exerciții | - |
| 12 | Tablouri și pointeri | expunere, dezbatere, exerciții | - |
| 13 | Structuri | expunere, dezbatere, exerciții | - |
| 14 | Exerciții recapitulative | expunere, dezbatere, exerciții | - |

Bibliografie*Intel® 64 and IA-32 Architectures Software Developer Manuals:*<http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html>**9. Coroborarea conținutului disciplinei cu așteptările reprezentanților comunității, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului**

Disciplina de față prezintă studenților elementele de bază ale unei arhitecturi de calcul. Aceste cunoștințe sunt esențiale pentru înțelegerea modului de lucru al calculatoarelor, indiferent de scopul pentru care sunt folosite; în particular, scrierea de programe performante nu este posibilă fără a cunoaște mecanismele și tehnicile predate aici.

10. Evaluare

| Tip activitate | 10.1 Criterii de evaluare | 10.2 Metode de evaluare | 10.3 Pondere în nota finală (%) |
|-------------------------|---|--|---------------------------------|
| 10.4 Curs | capacitatea de a aplica elementele teoretice la rezolvarea de probleme de natură practică | două teste scrise, fiecare din materia unei jumătăți de semestru | 40% 20% |
| 10.5 Seminar/ Laborator | capacitatea de a dezvolta programe, cu accent pus pe eficiența programării | test practic | 40% |

10.6 Standard minim de performanță

- cunoașterea la nivel minimal a conceptelor legate de hardware-ul sistemelor de calcul
- capacitatea de a dezvolta programe de complexitate redusă în limbajul de asamblare al familiei de procesoare Intel x86
- minimum 5 puncte la fiecare test; se acceptă mai puțin de 5 puncte (dar nu mai puțin de 2 puncte) la cel mult unul dintre teste
- nota finală se calculează ca media ponderată a celor trei teste; în cazul în care s-au obținut mai puțin de 5 puncte la unul dintre teste, rotunjirea se face prin trunchiere

Data completării

Titular de curs

Titular de seminar

Data avizării în departament

Director de departament