

FIȘA DISCIPLINEI*

1. Date despre program

| | |
|-----------------------------------|--|
| Instituția de învățământ superior | Universitatea Lucian Blaga din Sibiu |
| Facultatea | Facultatea de Științe |
| Departament | Departamentul de matematică și Informatică |
| Domeniul de studiu | Informatică |
| Ciclul de studii | Licență |
| Specializarea | Informatică |

2. Date despre disciplină

| | | | | |
|---|---|--------------|-----------|------------------|
| Denumirea disciplinei | Programare concurentă | | | |
| Codul cursului | Tipul cursului | An de studiu | Semestrul | Număr de credite |
| 380602O06045 | A | 3 | 6 | 5 |
| Tipul de evaluare | Categoriza formativă a disciplinei: DS (DF=fundamentală.; DD=domeniu; DS=specialitate; DC=complementară) | | | |
| Examen | Examen | | | |
| Titular activități curs | Conf. univ. dr. Florin Stoica | | | |
| Titular activități seminar / laborator/ proiect | Lect. univ. dr. Ralf Fabian | | | |

3. Timpul total estimat

| | | | | |
|---|---------|-----------|---------|------------------------|
| Extinderea disciplinei în planul de învățământ – număr de ore pe săptămână | | | | |
| Curs | Seminar | Laborator | Proiect | Total |
| 2 | - | 2 | - | 4 |
| Extinderea disciplinei în planul de învățământ – Total ore din planul de învățământ | | | | |
| Curs | Seminar | Laborator | Proiect | Total ($NOAD_{sem}$) |
| 24 | - | 24 | - | 48 |

| | | |
|--|--|------------|
| Distribuția fondului de timp pentru studiu individual | | Nr.ore |
| Studiul după manual, suport de curs, bibliografie și notițe | | 30 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren | | 10 |
| Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri | | 30 |
| Tutoriat: | | 5 |
| Examinări: | | 2 |
| Total ore alocate studiului individual ($NOSI_{sem}$) | | 77 |
| Total ore pe semestru ($NOAD_{sem} + NOSI_{sem}$) | | 125 |

4. Precondiții (acolo unde este cazul)

| | |
|---------------|--|
| De curriculum | Algoritmi și structuri de date, Programare procedurală |
| De competențe | |

5. Condiții (acolo unde este cazul)

| | |
|-----------------------------|--|
| De desfășurare a cursului | Sală de curs, dotată cu tablă, calculator, videoproiector și software (Adobe Reader, PowerPoint) |
| De desfășurare a sem/lab/pr | Sală de laborator dotată cu calculatoare desktop |

6. Competențe specifice acumulate

| | |
|-------------------------|--|
| Competențe profesionale | <ul style="list-style-type: none">• Potențialul de a valorifica la nivel de înțelegere competențele dobândite prin această disciplină.• Abilitatea transpunere a conceptelor teoretice și a tehnologiilor fundamentale de programare concurrentă în implementări practice.• Explicarea conceptelor și noțiunilor însușite.• Folosirea deprinderilor dobândite în punerea și rezolvarea de probleme practice.• Aptitudini de prezentare și promovare a propriilor rezultate obținute.• Utilizarea tehnicii de calcul pentru rezolvarea de probleme practice.• Studentul să dezvolte și implementeze software bazat pe biblioteci standard pentru dezvoltarea de aplicații concurente (POSIX threads, clasele JAVA, API Windows).• Însușirea aspectelor teoretice privind procesele și thread-urile, precum și mecanismele de sincronizare și coordonare a acestora.• Exploatarea facilităților de lucru cu procese și thread-uri oferite de către sistemele de operare Unix (Linux), Windows și de către platforma Java |
| Competențe transversale | <ul style="list-style-type: none">• Dezvoltarea atitudinii pozitive față de muncă și responsabilitate pentru propria pregătire profesională• Dezvoltarea spiritului de munca în echipa• Capacitatea de utilizare a mijloacelor moderne de documentare |

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

| | |
|-----------------------------------|---|
| Obiectivul general al disciplinei | <ul style="list-style-type: none">• Înțelegerea conceptelor și a tehnologiilor fundamentale de programare concurrentă..• Cunoașterea tehnicilor de programare concurrentă pe platforme multiple: UNIX, Windows, Java• Formarea deprinderilor necesare utilizării bibliotecilor standard pentru dezvoltarea de aplicații concurente (POSIX threads, clasele JAVA, API Windows).• Manifestarea interesului pentru cunoașterea rezultatelor teoretice și aplicative recente în domeniul programării concurente.• Adaptare la lucrul colaborativ, în echipă, pentru rezolvarea unor probleme din diverse domenii, prin implicarea în proiecte informatice în care se utilizează programarea concurrentă |
| Obiectivele specifice | <ul style="list-style-type: none">• Dobândirea cunoștințelor și aptitudinilor pentru aplicarea conceptelor teoretice de la curs în rezolvarea unor probleme concrete.• Implementarea unor algoritmi care abordează teme specifice programării concurente: excludere mutuală, resurse critice, secțiuni critice, sincronizarea accesului la resurse.• Evaluarea experimentală a performanțelor algoritmilor cu multiple fire de execuție executate în mod concurrent.• Exprimarea unui mod de gândire creativ în formularea unei soluții bazată pe programare concurrentă pentru probleme cu grad mediu de dificultate. |

8. Conținuturi

| Curs | | Nr. ore |
|------------------------|--|-----------|
| Curs 1-2 | <p>Nivelele prelucrărilor concurente</p> <ul style="list-style-type: none"> ▪ Procesări paralele și clasificarea Flynn ▪ Granularități ale paralelismului / concurenței ▪ Tehnica pipeline ▪ Procesoare vectoriale și sisteme cluster ▪ Paralelism și concurența la nivelul sistemului de operare ▪ Evaluare multiprocesor a expresiilor complexe ▪ Reorganizarea succesiunilor de atribuiri ▪ Paralelizare la nivel de cicluri for | 4 |
| Curs 3-4 | <p>Concepte abstracte utilizate în descrierea concurenței</p> <ul style="list-style-type: none"> ▪ Paradigme de programare nesecvențială ▪ Relația procese - thread-uri ▪ Scheme de specificare a programelor concurente ▪ Situații de excepție generate de concurență ▪ Mecanisme de control al concurenței, comunicare și sincronizare ▪ Mecanisme de control asincron sau parțial sincron ▪ Probleme specifice care se rezolvă cu ajutorul concurenței | 4 |
| Curs 5-6-7 | <p>Programare concurentă la nivel de proces</p> <ul style="list-style-type: none"> ▪ Procese Unix, Windows, Java ▪ Comunicarea prin pipe între procese ▪ Comunicarea între procese folosind mecanismul de memorie partajată ▪ Sincronizarea proceselor folosind semafoare ▪ Comunicarea prin cozi de mesaje | 6 |
| Curs 8-9-10 | <p>Programare concurentă la nivel de thread-uri</p> <ul style="list-style-type: none"> ▪ Caracteristici generale ▪ Exemple de probleme rezolvabile prin thread-uri ▪ Thread-uri pe platforme Unix: Posix și Solaris ▪ Thread-uri pe platforme Microsoft: Windows ▪ Thread-uri Java | 6 |
| Curs 11-12 | <p>Aplicații concurente complexe</p> <ul style="list-style-type: none"> ▪ Scheme de proiectare a programelor concurente ▪ Utilizări combinate: thread-uri, procese Unix, semnale ▪ Evaluarea unor performanțe ale programelor cu thread-uri | 4 |
| Total ore curs: | | 24 |
| Seminar/Laborator | | Nr. ore |
| Lab 1 | Procese Windows. Apeluri sistem pentru managementul proceselor Windows. | 2 |
| Lab 2 | Fire de execuție Windows. Apeluri sistem pentru managementul firelor de execuție Windows. | 2 |
| Lab 3 | Procese Linux. Apeluri sistem pentru managementul proceselor Linux. | 2 |
| Lab 4 | Fire de execuție Linux. Apeluri sistem pentru managementul firelor de execuție Linux. | 2 |
| Lab 5-6 | Mecanisme de comunicare inter-procese. Comunicația inter-procese prin pipe-uri. Comunicarea între procese folosind mecanismul de memorie partajată. Comunicarea prin cozi de mesaje. | 4 |
| Lab 7 | Mecanisme de sincronizare (mutex-uri, variabile de condiție, semnale). Aplicație vizuală multi-threading Windows | 2 |
| Lab 8-9 | Utilizarea thread-urilor în appleturi și servlet-uri Java | 4 |
| Lab 10-11 | Server Java concurent pentru chat. Client FTP noninteractiv. | 4 |

| | | |
|------------------------------------|---|----|
| Lab 12 | Testare finală/evaluare finală, prezentare proiecte | 2 |
| Total ore seminar/laborator | | 24 |

Metode de predare

| | | |
|---|--|--|
| <p>La curs se va folosi expunerea sistematică a cunoștințelor (deductivă, inductivă și formalizată, expuneri ppt); conversația frontală; conversația euristică, problematizare, studii de caz, modelarea.</p> <p>La laborator se va folosi explicația, exemplificarea, învățarea prin explorare și descoperire, verificarea, explicarea cauzelor, corectarea, stabilirea de relații funcționale, evaluarea, interpretarea, judecata critică, anticiparea, alte metode de învățare activă.</p> | | |
|---|--|--|

Bibliografie

| | |
|--------------------------------------|--|
| Referințe bibliografice recomandate | Stoica F., Cacovean L., Programare API, Ed. Psihomedica, 2009 |
| | F. M. Boian, C. Ferdean, R. Boian, R. Dragoș, Programare concurentă pe platforme Unix, Windows, Java, ISBN 973-650-072-1, Editura Alabastră - grupul Microinformatica, Cluj, 2002. |
| Referințe bibliografice suplimentare | Stoica F., Sisteme de operare, Ed. Universității „Lucian Blaga”, 2007. |
| | |
| | |

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

Prin utilizarea mecanismelor și tehnicilor de programare concurentă se dezvoltă aplicații software eficiente, care exploatează intensiv resursele de calcul puse la dispoziție. Dezvoltarea de aplicații moderne, responsive, este de mare actualitate în firmele de software.

Coroborarea conținutului disciplinei cu așteptările angajatorilor se realizează prin contacte periodice cu aceștia în vederea analizei și identificarea soluțiilor optime pentru rezolvarea problemei.

10. Evaluare

| Tip activitate | Criterii de evaluare | Metode de evaluare | Ponderea în nota finală | Obs. ** |
|----------------|--|-------------------------------------|-------------------------|---------|
| Curs | <ul style="list-style-type: none"> - corectitudinea și completitudinea cunoștințelor asimilate; - o înțelegere de ansamblu a importanței disciplinei studiate și a legăturii cu celelalte discipline de specialitate - coerența logică; - gradul de asimilare a limbajului de specialitate; - criteriile ce vizează aspectele atitudinale: interesul pentru studiul individual și dezvoltarea profesională. | Examen de semestru, evaluare scrisă | 50% | CEF |
| Laborator | <ul style="list-style-type: none"> - capacitatea de aplicare în practică a cunoștințelor teoretice; - criteriile ce vizează aspectele atitudinale: seriozitatea, interesul pentru studiul individual. | Activități aplicative | 10% | nCPE |

| | | | | |
|--|--|---------|-----|------|
| | capacitatea de a dezvolta aplicații concurente | Proiect | 40% | nCPE |
| Standard minim de performanță | | | | |
| <ul style="list-style-type: none"> • Înțelegerea conceptelor și a tehnologiilor fundamentale de programare concurentă • Cunoașterea tehnicilor de programare concurentă pe platforme multiple: Windows, Java | | | | |

(*) Fișa disciplinei cuprinde componente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

(**) CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală;

Data completării: 26.09.2020

Data avizării în Departament:.....

| | Grad didactic, titlul, prenume, numele | Semnătura |
|-------------------------|--|-----------|
| Titular disciplină | Conf. univ. dr. Florin Stoica | |
| Director de departament | Prof.univ.dr. Mugur Acu | |