

## FIȘA DISCIPLINEI

### 1. Date despre program

|  |  |
|--|--|
| 1.1. Instituția de învățământ superior | Universitatea „Tibiscus” din Timișoara   |
| 1.2. Facultatea                        | Calculatoare și Informatică Aplicată   |
| 1.3. Departamentul                     | Informatică  |
| 1.4. Domeniul de studii                | Informatică  |
| 1.5. Ciclul de studii                  | Licență  |
| 1.6. Programul de studii/Calificarea   | Informatică / Asistent de cercetare în informatică - 214918; Proiectant sisteme informatice - 25110; Analist - 251201; Administrator baze de date - 252101; Administrator de rețea de calculatoare - 252301; Profesor în învățământul gimnazial - 233002; Programator - 251202 |

### 2. Date despre disciplină

|   |   |                |   |                        |   |                          |    |
|---|---|----------------|---|------------------------|---|--------------------------|----|
| 2.1. Denumirea/codul disciplinei        | <b>PROGRAMARE CONCURRENTĂ ȘI DISTRIBUITĂ (PCD) - (OPT.1)   LIN231</b> |                |   |                        |   |                          |    |
| 2.2. Titularul activității de curs      | Conf.dr.ing. Laurențiu - Dan Lacrămă                                  |                |   |                        |   |                          |    |
| 2.3. Titularul activității de laborator | Conf.dr.ing. Laurențiu - Dan Lacrămă                                  |                |   |                        |   |                          |    |
| 2.4. Anul de studiu                     | 2   | 2.5. Semestrul | 1 | 2.6. Tipul de evaluare | E | 2.7. Regimul disciplinei | DS |

### 3. Timpul total estimat

|  |    |                    |    |                |            |
|--|----|--------------------|----|----------------|------------|
| 3.1. Numărul de ore pe săptămână   | 3  | din care 3.2. curs | 1  | 3.3. laborator | 2          |
| 3.4. Total ore din planul de învățământ  | 42 | din care 3.5. curs | 14 | 3.6. laborator | 28         |
| <b>Distribuția fondului de timp</b>  |    |                    |    |                | <b>ore</b> |
| Studiul după manual, suport de curs, bibliografie și notițe                      |    |                    |    |                | 18         |
| Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate |    |                    |    |                | 16         |
| Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri            |    |                    |    |                | 24         |
| Tutoriat   |    |                    |    |                | 6          |
| Examinări  |    |                    |    |                | 2          |
| Alte activități  |    |                    |    |                |            |
| <b>3.7. Total ore studiu individual</b>  |    |                    |    |                | <b>58</b>  |
| <b>3.8. Total ore pe semestru</b>  |    |                    |    |                | <b>100</b> |
| <b>3.9. Numărul de credite</b>   |    |                    |    |                | <b>4</b>   |

### 4. Precondiții (acolo unde este cazul)

|                    |  |
|--------------------|--|
| 4.1. de curriculum | -  |
| 4.2. de competențe | Sisteme de operare, Sisteme de operare distribuite, Programare procedurală, Tehnici avansate de programare |

### 5. Condiții (acolo unde este cazul)

|   |                            |
|---|----------------------------|
| 5.1. de desfășurare a cursului                  | Online: Google Drive, Zoom |
| 5.2. de desfășurare a seminarului/laboratorului | Online: Google Drive, Zoom |

### 6. Competențe specifice acumulate

|                              |   |
|------------------------------|---|
| 6.1. Competențe profesionale | C1. Concepte abstracte utilizate în descrierea concurenței și distribuției (paradigme de programare nesecvențială, noțiunile de proces și thread, 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, sisteme distribuite vs. sisteme paralele, programe |
|------------------------------|---|



|                              |  |
|------------------------------|--|
|                              | <p>distribuite, elemente de programare distribuită).</p> <p>C2. Programare concurentă la nivel de proces (proces Unix, comunicarea prin pipe între procese, comunicarea între procese folosind mecanismul de memorie partajată, sincronizarea proceselor folosind semafoare).</p> <p>C3. Programare concurentă la nivel de thread-uri (caracteristici generale, exemple de probleme rezolvabile prin thread-uri pe platforma Unix).</p> <p>C4. Programare distribuită la nivel de proces (socket-uri Berkeley, exemple rezolvate de servere TCP și UDP, Socluri și Semnale. Operații de Intrare / ieșire asincrone pe platforma Unix).</p> |
| 6.2. Competențe transversale | -  |

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

|  |  |
|--|--|
| 7.1. Obiectivul general al disciplinei | <ul style="list-style-type: none"> <li>Furnizarea de cunoștințe de bază despre tehnologiile fundamentale de programare concurentă și distribuită: aspectele teoretice și practice privind procesele și thread-urile, precum și instrumentele de coordonare a acestora.</li> <li>Dezvoltarea de aplicații software conform stilului de lucru concurent și distribuit.</li> </ul>  |
| 7.2. Obiectivele specifice             | <ul style="list-style-type: none"> <li>Înșușirea principalelor concepte care operează în cadrul operează în cadrul activității de proiectare software de aplicații concurente și/sau distribuite.</li> <li>Înșușirea cunoștințelor și deprinderilor practice care au în vedere facilitățile de lucru cu procese și thread-uri oferite de către sistemul de operare Unix (Oracle Solaris 11)</li> <li>Dobândirea unor cunoștințe pentru abordarea cu succes a proiectelor software complexe.</li> </ul> |

## 8. Conținuturi

| 8.1. Curs  | Metode de predare  | Observații |
|--|--|------------|
| 1. Procese și thread-uri, Multiprogramare, Paradigme de programare         | Expunerea interactivă, problematizarea, conversația euristică, documentarea pe web, exemplificarea | 2 ore      |
| 2. Programarea concurentă și distribuită                                   |  | 4 ore      |
| 2.1. Paradigme de programare nesecvențială                                 |  |            |
| 2.2. Programare paralelă   |  |            |
| 2.3. Programare concurentă   |  |            |
| 2.4. Scheme de specificare a programelor concurente                        |  |            |
| 2.5. Programare distribuită  |  |            |
| 3. Programare concurentă și distribuită în UNIX                            | Expunerea interactivă, problematizarea, conversația euristică, documentarea pe web, exemplificarea | 8 ore      |
| 3.1. Procese UNIX  |  |            |
| 3.2. Thread-uri UNIX   |  |            |
| 3.3. Comunicarea prin pipe între procese                                   |  |            |
| 3.4. Unirea și detașarea thread-urilor                                     |  |            |
| 3.5. Sincronizarea thread-uri folosind mecanismul de excludere mutuală     |  |            |
| 3.6. Sincronizarea thread-uri folosind mecanismul de variabilă de condiție |  |            |
| 3.7. Sincronizarea proceselor folosind semafoare                           |  |            |
| 3.8. Blocarea accesului la fișiere utilizând semafoare                     |  |            |
| System V   |  |            |
| 3.9. Transferuri cu conexiune  |  |            |
| 3.10. Transferuri cu conexiune- protocol TCP                               |  |            |
| 3.11. Exemplu demonstrativ: Client/Server TCP                              |  |            |
| 3.12. Transferurile fără conexiune   |  |            |
| 3.13. Transferurile fără conexiune- protocol UDP                           |  |            |
| 3.14. Exemplu demonstrativ: Client/Server UDP                              |  |            |
| 3.15. Socluri și Semnale. Operații de Intrare/Ieșire asincrone.            |  |            |

### Bibliografie

- Virgiliu Streian și Lucian Luca: Sistemul de operare UNIX, Editura Mirton Timișoara 1998
- Lucian Luca: Sisteme de operare distribuite, Editura Mirton Timișoara 1998
- Virgiliu Streian: Dezvoltarea de aplicații UNIX, masterat- Curs intern, FCIA 2011
- Virgiliu Streian: Proiectarea unui Token Ring, FCIA, Anale. Seria informatică. Vol .V Fasc.I – 2007
- BACON J., Concurrent Systems, Addison-Wesley, England, 1998.
- BARRY A., Concurrent Programming, <http://www.csm.uwe.ac.uk/personal/am-barry/Q2H611/concprog.html>
- BOIAN F.M., FERDEAN C. M., BOIAN R.F. DRAGOS R.C. Programare concurenta pe platforme Unix, Windows, Editura Albastra - grupul Microinformatica, Cluj, 2002.
- FLYNN M. J., Some Computer Organizations and their Effectiveness, IEEE Transactions on Computers, C-21, 1972
- <http://www.rasfoiesc.com/educatie/informatica/Programare-concurenta18.php>
- <http://www.rasfoiesc.com/educatie/informatica/Programare-concurenta-Relatia-28.php>



| 8.2. Seminar/laborator  | Metode de predare/învățare  | Observații          |
|---|---|---------------------|
| 1. Măsurarea timpului real și a timpului virtual (timp CPU) a unui proces.  | Exercițiul, discuțiile și dezbaterile, modelarea, proiectul, lucrul în grup organizat | 1 săptămână – 2 ore |
| 2. Crearea de procese într-o structură ierarhizată.   |   | 1 săptămână – 2 ore |
| 3. Crearea și terminarea unui thread.   |   | 1 săptămână – 2 ore |
| 4. Conectarea a două procese printr-un PIPE (buffer de conectare) cu redirecționarea fișierelor standard de intrare respectiv ieșire. |   | 2 săptămâni – 4 ore |
| 5. Sincronizare thread-uri prin excludere mutuală.  |   | 1 săptămână – 2 ore |
| 6. Sincronizarea thread-urilor pe variabile de condiție.  |   | 1 săptămână – 2 ore |
| 7. Gestiunea stivei thread-urilor.  |   | 1 săptămână – 2 ore |
| 8. Cronometre de interval.  |   | 1 săptămână – 2 ore |
| 9. Cronometre realtime - Cronometre de interval POSIX.1b.   |   | 1 săptămână – 2 ore |
| 10. Așteptarea după un semnal.  |   | 1 săptămână – 2 ore |
| 11. Server TCP cu conexiune.  |   | 1 săptămână – 2 ore |
| 12. Client / server UDP fără conexiune.   |   | 2 săptămâni – 4 ore |
| <b>Bibliografie</b>   |   |                     |
| - Streian, Virgiliu - curs intern <i>Programare concurentă și distribuită</i>   |   |                     |

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

Conținutul disciplinei corespunde curriculei din alte centre universitare, din țară sau Uniunea Europeană.  
 Conținuturile practice (lucrări de laborator) corespund cerințelor de pe piața muncii locală.

### 10. Evaluare

| Tip de activitate  | 10.1. Criterii de evaluare   | 10.2. Metode de evaluare   | 10.3. Pondere din nota finală |
|--|--|--|-------------------------------|
| 10.4. Curs   | Evaluarea are în vedere următoarele categorii de cunoștințe: <ul style="list-style-type: none"> <li>• Cunoștințe generale, evaluate printr-un test cuprinzând întrebări cu variante multiple de răspuns.</li> <li>• Cunoștințe de detaliu, evaluate printr-un test cuprinzând întrebări cu variante multiple de răspuns orientate spre noțiunile cheie predate.</li> </ul> | Examinare scrisă.  | 50%                           |
| 10.5. Seminar / laborator  | Evaluarea are în vedere următoarele categorii de cunoștințe: <ul style="list-style-type: none"> <li>• Cunoștințe generale: utilizarea limbajului C pentru dezvoltare de aplicații.</li> <li>• Cunoștințe de detaliu: dezvoltarea de aplicații corespunzătoare fiecărei teme de laborator.</li> </ul>   | Evaluarea activității la laborator; Participarea activă la activitățile de laborator; Examinare scrisă printr-un test final cu întrebări cu variante multiple de răspuns orientate spre recunoașterea unor secvențe de program asemănătoare de la laborator. | 50%                           |
| 10.6. Standard minim de performanță  |  |  |                               |
| <i>Examinare scrisă:</i><br>Pentru nota 5 este necesar un număr de 5 răspunsuri corecte din 10 posibile la un test de tip grilă<br>Pentru nota 10 este necesar un număr de 10 răspunsuri corecte din 10 posibile la un test de tip grilă<br><i>Probe practice și activitate de laborator:</i><br>Pentru nota 5 este necesară obținerea unui nivel superior (min. 60%) pentru cunoștințe generale, precum și a unui nivel minim de înțelegere și utilizare a cunoștințelor de detaliu prezentate anterior.<br>Pentru nota 10 este necesară obținerea unui nivel superior (min. 80%) pentru cunoștințe de detaliu prezentate anterior. |  |  |                               |

Data completării

28.09.2020

Semnătura titularului de curs

.....

Semnătura titularului de laborator

.....

Data avizării în departament

05.10.2020



Semnătura directorului de departament

.....