|  |  |  |  |
| --- | --- | --- | --- |
|  | **FIŞA DISCIPLINEI/MODULULUI** | **Cod: F.O. 006** | |
| **STRUCTURI DE DATE ŞI ALGORITMI** | **Ediţia** | **1** |
| **Revizia** | **1** |
| **Pagina** | **1 / 4** |

**FIŞA DISCIPLINEI/ MODULULUI**

MD-2045, CHIŞINĂU, STR. STUDENȚILOR, 9/7, TEL: 022 50-99-01 | FAX: 022 50-99-05, [www.utm.md](http://www.utm.md/)

STRUCTURI DE DATE ŞI ALGORITMI

1. **Date despre disciplină/modul**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Facultatea** | Calculatoare, Informatică și Microelectronică | | | | |
| **Departamentul** | Informatică şi Ingineria Sistemelor | | | | |
| **Ciclul de studii** | Studii superioare de licenţă, ciclul I | | | | |
| **Programul de studii** | 0612.3 Ştiinţa Datelor | | | | |
| **Anul de studii** | **Semestrul** | **Tip de**  **evaluare** | **Categoria**  **formativă** | **Categoria de**  **opţionalitate** | **Credite**  **ECTS** |
| I (învăţământ cu frecvenţă); | 2 | E | F – unitate de curs de  fundamentală | O - unitate de curs  obligatorie | 5 |

1. **Timpul total estimat**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Total ore în planul de  învăţământ | Din care | | | | |
| Ore auditoriale | | Lucrul individual | | |
| Curs | Laborator/seminar | Proiect de  an | Studiul materialului  teoretic | Pregătire aplicaţii |
| 180 | 30 | 30/30 | 30 | 30 | 30 |

1. **Precondiţii de acces la disciplină/modul**

|  |  |
| --- | --- |
| Conform planului de  învăţământ | Programarea calculatoarelor, Matematici speciale |
| Conform competenţelor | Cunoştinţe şi abilităţi de concepere şi elaborare a algoritmilor şi programelor în  C pentru rezolvarea problemelor la calculator |

1. **Condiţii de desfăşurare a procesului educaţional pentru**

|  |  |
| --- | --- |
| Curs | Pentru prezentarea materialului teoretic în sala de curs este nevoie de tablă, proiector şi  calculator. |
| Laborator/seminar | Studenţii vor perfecta rapoarte conform condiţiilor formulate în indicaţiile metodice. Termenul de susţinere a lucrării de laborator – o săptămână după finalizarea acesteia. Pentru prezentarea cu întârziere a lucrării, aceasta se depunctează cu 1pct./săptămână  de întârziere. |

1. **Competenţe specifice acumulate**

|  |  |  |
| --- | --- | --- |
| Competenţe profesionale | CP2. Proiectarea și dezvolta rea aplicațiilor (B.1)  Programe/module software adecvate.  Proiectarea funcțională și tehnică.  Tehnologiile de ultimă oră.  Limbaje de programare.  Mediul de dezvoltare integrat (IDE - integrated develop-ment environment).  Problemele legate de drepturile de proprietate intelectuală (IPR).  Tehnologia de modelare tehnică și limbaje.  Probleme de securitate.  Efectuează și evaluează rezultatele testelor în funcție de specificațiile produsului  Gestionează și garantează nivel ridicat de calitate şi de coeziune.  Efectuează și evaluează rezultatele testului în mediul client sau mediul țintă.  Colaborează cu echipa de dezvoltare și cu designerii de aplicații.  CP3. Integrarea componentelor (B.2)  Componente/module hardware/software, indiferent dacă sunt vechi, existente sau noi.  Impactul integrării unui sistem asupra organizației sau a sistemului existent.  Tehnici de interfațare între module, sisteme și componente.  Tehnici de testare a integrării.  Instrumentele de dezvoltare (ex. mediul de dezvoltare, gestionare, control al modificărilor și accesul la codul sursă).  Bune practici de design  Măsoară performanța sistemului înainte, în timpul și după integrarea sistemului  Identifică și înregistrează activitățile, problemele și măsurile corective legate de întreținere.  P8. Marketing digital (D.12)  Strategii de marketing.  Optimizarea motoarelor de căutare.  Marketingul legat de instrumente mobile (de ex. Pay Per Click).  Marketing legat de media social.  Display marketing.  Probleme/cerințe legale.  Înțelege modul în care tehnologia web poate fi utilizată în scopuri de marketing.  Înțelege marketingul orientat către utilizator.  Înțelege problemele legate de mediile online. | |
| Competenţe transversale | CT1. Aplicarea, în contextul respectării legislaţiei, a drepturilor de proprietate intelectuala (inclusiv transfer tehnologic), a metodologiei de certificare a produselor, a principiilor, normelor şi valorilor codului de etică profesională în cadrul propriei strategii de muncă  riguroasă, eficientă şi responsabilă. |
| CT2. Identificarea rolurilor şi responsabilităţilor într-o echipă şi aplicarea de tehnici de  relaţionare şi muncă eficientă în cadrul echipei. |
| CT3. Identificarea oportunităţilor de formare continuă şi valorificarea eficientă a resurselor  şi tehnicilor de învăţare pentru propria dezvoltare. |

|  |  |
| --- | --- |
| Obiectivul general | Însuşirea profundă a structurilor de date şi fişiere, metodelor de sortare a datelor,  aplicaţiilor dinamice şi a algoritmilor de căutare a soluţiilor optimale. |
| Obiectivele specifice | Să înţeleagă şi să descrie algoritmii de prelucrare a structurilor elaborate. Să elaboreze funcțiile necesare pentru prelucrarea fișierelor.  Să aplice corect algoritmii și procedeele de operare în rezolvarea problemelor cu  aplicații dinamice. |

1. **Obiectivele disciplinei/modulului**
2. **Conţinutul disciplinei/modulului**

|  |  |  |
| --- | --- | --- |
| **Tematica activităţilor didactice** | **Numărul de ore** | |
| **învăţământ cu frecvenţă** | **învăţământ cu frecvenţă**  **redusă** |
| **Tematica cursurilor** | | |
| **T1** Sortarea datelor. Algoritmi de sortare şi analiza performanţelor. | 4 |  |
| **T2** Structuri de date *struct*, *union*, cu câmpuri de *biţi*. Declararea, iniţializarea,  citirea şi afișarea structurilor. | 4 |  |
| **T3** Tipul de date FILE. Fişiere de tip logic şi fizic. Funcţii predefinite pentru date de tip FILE: fopen(), fread(), fwrite(), fprintf(), fscanf(), fseek(), feof().  Operații cu fișiere: creare, afişare, adăugare, corectare, sortare. Acces secvenţial şi direct la fişiere. | 6 |  |

|  |  |  |
| --- | --- | --- |
| **T4** Structuri dinamice de date. Alocare de memorie. Funcţii predefinite. Aplicaţii dinamice: stiva, şir de aşteptare, listă liniară, lista înlănţuită, listă bidirecţională. Arbori. Arbori binari de căutare. Proceduri de operare cu  aplicaţii dinamice: creare, parcurgere, afișare, modificare. | 6 |  |
| **T5** Algoritmi şi metode de căutare optimală a soluţiilor. Clasificarea  algoritmilor. Avantaje şi dezavantaje ale algoritmilor din perspectiva: universalității, capacității de memorie utilizate și a vitezei de execuție. | 2 |  |
| **T6** Algoritmii care garantează soluția optimală: *Backtracking , Programarea*  *dinamică, Divde et Impera, Branch and Bound.* | 4 |  |
| **T7** Algorimii *Greedy, euristici, probabilistici, genetici*. Analiza algoritmilor. | 4 |  |
| **Total curs:** | **30** |  |
| **Tematica lucrărilor practice/seminarelor** | | |
| **LP1** Sortarea datelor. Algoritmi de sortare şi analiza performanţelor | 2 |  |
| **LP2** Structuri de date *struct*, *union*, cu câmpuri de *biţi*. Declararea,  iniţializarea, citirea şi afişarea structurilor. | 2 |  |
| **LP3** Tipul de date FILE. Fişiere de tip logic şi fizic. Funcţii predefinite pentru date de tip FILE. Operaţii cu fişiere: creare, afişare, adăugare, corectare,  sortare. Acces secvenţial şi direct la fişiere. | 3 |  |
| **LP4, LP5** Structuri dinamice de date. Alocare de memorie. Funcţii predefinite. Aplicații dinamice: stiva, şir de aşteptare, listă liniară, lista înlănțuită, listă bidirecţională. Arbori. Arbori binari de căutare. | 3 |  |
| **LP6** Algoritmi şi metode de căutare optimală a soluţiilor. Algoritmii care garantează soluția optimală: *Backtracking , Programarea dinamică, Divde et*  *Impera, Branch and Bound.* | 3 |  |
| **LP7** Algorimii *Greedy, euristici, probabilistici, genetici*. Analiza algoritmilor. | 2 |  |
| **Total lucrări practice/seminare:** | **15** |  |
| **Tematica lucrărilor de laborator** | | |
| **LL1** Sortarea datelor. Algoritmi de sortare şi analiza performanţelor. | 4 |  |
| **LL2** Structuri de date *struct*, *union*, cu câmpuri de *biţi*. Declararea,  iniţializarea, citirea şi afişarea structurilor. | 4 |  |
| **LL3** Tipul de date FILE. Fişiere de tip logic şi fizic. Funcţii predefinite pentru date de tip FILE. Operaţii cu fişiere: creare, afişare, adăugare, corectare,  sortare. Acces secvenţial şi direct la fişiere. | 6 |  |
| **LL4, LL5** Structuri dinamice de date. Alocare de memorie. Funcţii predefinite. Aplicații dinamice: stiva, şir de aşteptare, listă liniară, lista înlănțuită, listă bidirecţională. Arbori. Arbori binari de căutare. | 6 |  |
| **LL6** Algoritmi şi metode de căutare optimală a soluţiilor. Algoritmii care garantează soluţia optimală: *Backtracking , Programarea dinamică, Divide et*  *Impera, Branch and Bound.* | 6 |  |
| **LL7** Algorimii *Greedy, euristici, probabilistici, genetici*. Analiza algoritmilor. | 4 |  |
| **Total lucrări de laborator:** | **30** |  |

1. **Referinţe bibliografice**

|  |  |
| --- | --- |
| Principale | 1. Florian Moraru. Structuri de Date. Disponibil on-line: <http://andrei.clubcisco.ro/cursuri/f/f-sym/1sd/sd2010A4.pdf> 2. Jack Straub. C Programming: Data Structures and Algorithms. 167 p. Disponibil on-line: <https://faculty.washington.edu/jstraub/dsa/Master_2_7a.pdf> 3. Logofătu Doina. Bazele programării în C : aplicații. Doina Logofătu. Iaşi : Polirom, 2006. 406 p. ISBN 973-46-0219-5. 4. Ștefănescu Diana. Programarea in limbajele C/C++. Noțiuni de bază. 400p. București: Matrix Rom, 2002. ISBN / ISSN 973-685-475-2. 5. Tudor Liviana. Bazele programării in C. 240 p. ISBN / ISSN 978-973-755-644-   8. Editura: Matrixrom. București, 2010. |
| Suplimentare | 1. Buzurniuc, Şt. Iniţiere în limbajul C. Evrica. Chişinău, 2004. 2. Cechez, Em., Şerban, M. Programarea în limbajul C/C++. Editura: POLIROM. București, 2005. 3. Claude Delannoy Programmer en langage C. Ucors et exercices corriges. EYROLLES, Paris, 2002. 4. Ritchie, D., Kernighan, B. W.. C Programming Language. Publisher: Prentice Hall; 2nd edition 1988. 217 p. ISBN / ISSN: 9780131103627. Disponibil on-line: [http://www2.cs.uregina.ca/~hilder/cs833/Other%20Reference%20Materials/The%20](http://www2.cs.uregina.ca/~hilder/cs833/Other%20Reference%20Materials/The) C%20Programming%20Language.pdf 5. Negrescu Liviu. Limbajul C. Vol I, Vol II, Editura albastră. Cluj-Napoca, 1999. 6. Pătruţ, B. Aplicaţii în C şi C++. Bogdan Pătruţ. Bucureşti: Teora, 1998. 167 p. ISBN 973-601-760-5. 7. Sedgewick R.. Algorithmes en language C. DUNOD, Paris, 2001. |

1. **Evaluare**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Periodică | | Curentă | Studiu individual | Proiect/teză | Examen |
| EP 1 | EP 2 |
| 10% | 10% | 10% | --- | 40% | 60% |
| Standard minim de performanţă  Prezența la lecții; activitatea și calitatea pregătirii la / pentru prelegeri şi lucrări de laborator; Obținerea notei minime de „5” la fiecare dintre atestări şi lucrări de laborator;  Obținerea notei minime de „5” la proiectul de an (se aplică după caz);  Demonstrarea în lucrarea de examinare finală a cunoașterii tehnicilor de programare aplicate asupra structurilor de date prin implementarea algoritmilor studiați la rezolvarea problemelor în limbajul de  programare C. | | | | | |