**fişa disciplinei**

**1. Date despre program**

|  |  |
| --- | --- |
| 1.1 Instituţia de învăţământ superior | Universitatea Babeș-Bolyai Cluj-Napoca |
| 1.2 Facultatea | Matematica și Informatică |
| 1.3 Departamentul | Informatică |
| 1.4 Domeniul de studii | Informatică |
| 1.5 Ciclul de studii | Licență |
| 1.6 Programul de studiu / Calificarea | Informatică (în limba germană) |

**2. Date despre disciplină**

|  |  |
| --- | --- |
| 2.1 Denumirea disciplinei (de)(ro)(en) | Formale Sprachen und KompiliertechnikenLimbaje formale și tehnici de compilareFormal languages and compilers |
| 2.2 Titularul activităţilor de curs | Prof. dr. Klaus Dohmen |
| 2.3 Titularul activităţilor de seminar | Prof. dr. Klaus Dohmen |
| 2.4 Anul de studiu | 3 | 2.5 Semestrul | 1 | 2.6. Tipul de evaluare | E | 2.7 Regimul disciplinei | obligatorie |
| 2.8 Codul disciplinei | MLG5023 |  |

**3. Timpul total estimat** (ore pe semestru al activităţilor didactice)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3.1 Număr de ore pe săptămână | 6 | Din care: 3.2 curs | 2 | 3.3 seminar/laborator | 4 |
| 3.4 Total ore din planul de învăţământ | 84 | Din care: 3.5 curs | 28 | 3.6 seminar/laborator | 56 |
| Distribuţia fondului de timp: | ore |
| Studiul după manual, suport de curs, bibliografie şi notiţe | 10 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren | 10 |
| Pregătire seminarii/laboratoare, teme, referate, portofolii şi eseuri | 10 |
| Tutoriat | 8 |
| Examinări  | 3 |
| Alte activităţi: .................. | - |
| 3.7 Total ore studiu individual | 41 |
| 3.8 Total ore pe semestru | 125 |
| 3.9 Numărul de credite | 5 |

**4. Precondiţii** (acolo unde este cazul)

|  |  |
| --- | --- |
| * 1. de curriculum
 | * structuri de date si algoritmi
 |
| 4.2 de competenţe | * abilitati de programare (nivel mediu)
 |

**5. Condiţii** (acolo unde este cazul)

|  |  |
| --- | --- |
| 5.1 De desfăşurare a cursului |  |
| 5.2 De desfăşurare a seminarului/laboratorului | - laborator dotat cu calculatoare; mediu de dezvoltare pentru limbaje de programare nivel înalt. |

**6. Competenţele specifice acumulate**

|  |  |
| --- | --- |
| **Competențe profesionale** | C 4.1 Definirea conceptelor şi principiilor de bază ale informaticii, precum şi a teoriilor şi modelelor matematice C 4.2 Interpretarea de modele matematice şi informatice (formale) C 4.3 Identificarea modelelor si metodelor adecvate pentru rezolvarea unor probleme reale C 4.4 Utilizarea simulării pentru studiul comportamentului modelelor realizate si evaluarea performantelor C 4.5 Încorporarea de modele formale în aplicaţii specifice din diverse domenii |
| **Competențe transversale** | CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile faţă de domeniul didactic-ştiinţific, pentru valorificarea creativă a propriului potenţial, cu respectarea principiilor şi a normelor de etică profesionalăCT3 Utilizarea unor metode şi tehnici eficiente de învăţare, informare, cercetare şi dezvoltare a capacităţilor de valorificare a cunoştinţelor, de adaptare la cerinţele unei societăţi dinamice și de comunicare în limba română și într-o limbă de circulație internațională |

**7. Obiectivele disciplinei** (reieşind din grila competenţelor acumulate)

|  |  |
| --- | --- |
| 7.1 Obiectivul general al disciplinei | - să fie capabili să înțeleagă metodele de proiectare a compilatoarelor- îmbunătățirea abilităților de programare |
| 7.2 Obiectivele specifice | - cunoștințe despre back-end-ul unui compilator- înțelegerea conceptelor limbajelor formale și dezvoltarea abilităților de a lucra cu ele- înțelegerea și capacitatea de a lucra cu concepte specifice compilatoarelor (analiza lexicală, analiza sintactică) |

**8. Conţinuturi**

|  |  |  |
| --- | --- | --- |
| 8.1 Curs | Metode de predare | Observaţii |
| 1. Prezentarea generală a problemelor proiectării și implementării unui compilator. Gramatici și limbaje | Expunere: descrierea, explicația, exemplificarea |  |
| 2. Analiza lexicală | Expunere: descrierea, explicația, exemplificarea |  |
| 3. Gramatici regulare. Automate finite; determinism. Expresii regulare. Introducere: definiții, exemple | Expunere: descrierea, explicația, exemplificarea |  |
| 4. Gramatici independente de context (GIC). Introducere: definiții, exemple | Expunere: descrierea, explicația, exemplificarea |  |
| 5. Automate push-down (APD).Construcția unui APD echivalent cu o GIC data. | Expunere: descrierea, explicația, exemplificarea, studiu de caz |  |
| 6. Gramatici speciale - gramatici LL(k)Analiza LL(1) | Expunere: descrierea, explicația, exemplificarea, studiu de caz |  |
| 7. Gramatici speciale - gramatici LR(k)Analiza LR(0) | Expunere: descrierea, explicația, exemplificarea, studiu de caz |  |
| 8. Gramatici speciale - gramatici LR(k)Analiza SLR, LR(1), LALR | Expunere: descrierea, explicația, exemplificarea, studiu de caz |  |
| 9. Structura unui compilator. Unde se folosesc gramatici / automate | Expunere: descrierea, explicația, exemplificarea |  |
| 10. Generator de analiză lexicală lex/flex. Generator de analiză sintactică yacc/bison | Expunere: descrierea, explicația, exemplificarea |  |
| 11. Demonstrații, construcții și aplicații. Echivalența dintre automatele finite și gramaticile regulare. Echivalența dintre AF și expresiile regulare. Echivalența unor tipuri de gramatici independente de context. | Expunere: descrierea, explicația, exemplificarea, demonstrarea |  |
| 12 Demonstrații, construcții și aplicații. Proprietăți ale limbajelor regulare (proprietăți de inchidere, lema de pompare). Proprietăți ale limbajelor independente de context (leme de pompare). | Expunere: descrierea, explicația, exemplificarea, studiu de caz, demonstrarea |  |
| 13. Analiza semantică, forme intermediare, generarea codului intermediar, optimizarea codului intermediar. Introducere | Expunere: descrierea, explicația, exemplificarea |  |
| 14. Aplicații ale mecanismelor formale studiate în analiza semantică și generarea codului intermediar | Expunere: descrierea, explicația, exemplificarea |  |
| **Bibliografie în limba germană**[1] C. WAGENKNECHT, HIELSCHER M., Formale Sprachen, abstrakte Automaten und Compiler, Vieweg Teubner, 2009.[2] ASTEROTH, A., BAIER, C., Theoretische Informatik, eine Einführung in Berechnbarkeit, Komplexität und formale Sprachen, Pearson Studium, 2002.[3] HROMKOVIC, J., Theoretische Informatik, Formale Sprachen, Berechenbarkeit, Komplexitätstheorie, Algorithmik, Kommunikation und Kryptographie, Vieweg Teubner, 2011.**Bibliografie în alte limbi**[1] K.D. COOPER, L. TORCZON - Engineering a Compiler, Elsevier Science & Technology, 2011.[2] A.V. AHO, D.J. ULLMAN - Principles of compiler design, Addison-Wesley, 1978. |
| **8.2 Seminar**  | Metode de predare | Observaţii |
| 1-2. Specificarea unui limbaj de programare. Notatia BNF | Explicația, dialogul, studiu de caz |  |
| 3-4. Gramatici, limbaje generate de o gramatica, gramatica corespunzatoare unui limbaj | Dialogul, dezbaterea, studiul de caz, exemplificarea, demonstratia |  |
| 5-6. Automate finite: limbaj generat de un automat finit, automat finit corespunzator unui limbaj | Dialogul, dezbaterea, studiul de caz, exemplificarea,  |  |
| 7-8. Gramatici independente de context. Analiza sintactica LL(1) | Dialogul, dezbaterea, studiul de caz, exemplificarea,  |  |
| 9-10. Analiza sintactica LR(k) | Dialogul, dezbaterea, studiul de caz, exemplificarea |  |
| 11-12. Proprietati ale limbajelor. Demostratii si aplicatii.  | Dialogul, dezbaterea, studiul de caz, exemplificarea, demonstratia |  |
| 13-14. APD. Gramatici de tip 1, 2 si 3 (ierahia Chomsky) | Dialogul, dezbaterea, studiul de caz, exemplificarea |  |
| **Bibliografie în limba germană**[1] C. WAGENKNECHT, HIELSCHER M., Formale Sprachen, abstrakte Automaten und Compiler, Vieweg Teubner, 2009.[2] ASTEROTH, A., BAIER, C., Theoretische Informatik, eine Einführung in Berechnbarkeit, Komplexität und formale Sprachen, Pearson Studium, 2002.[3] HROMKOVIC, J., Theoretische Informatik, Formale Sprachen, Berechenbarkeit, Komplexitätstheorie, Algorithmik, Kommunikation und Kryptographie, Vieweg Teubner, 2011.**Bibliografie în alte limbi**[1] K.D. COOPER, L. TORCZON - Engineering a Compiler, Elsevier Science & Technology, 2011.[2] A.V. AHO, D.J. ULLMAN - Principles of compiler design, Addison-Wesley, 1978. |
| **8.3 Laborator** | Metode de predare | Observaţii |
| 1. Tema 1: Specificarea unui mini-limbaj de programare și implementarea unui analizor lexical

1.1 specificarea mini-limbajului de programare | Explicația, dialogul, studiu de caz |  |
| 1. Tema 1: Specificarea uni mini-limbaj de programare și implementarea unui analizor lexical

1.2 implemetarea funcțiilor principale ale analizorului | Explicația, dialogul, studiu de caz  |  |
| 1. Tema 1: Specificarea uni mini-limbaj de programare și implementarea unui analizor lexical

1.3 organizarea tabelei de simboluri | Explicația, dialogul, studiu de caz  |  |
| 1. Tema 1: Specificarea uni mini-limbaj de programare și implementarea unui analizor lexical

1.4 programul principal, testare + predare | Discutarea datelor de test, evaluare |  |
| 1. Tema 2: Automate finite

2.1 verificarea acceptării unei secvențealegerea structurilor de date și arhitectura aplicației | Explicația, dialogul, studiu de caz |  |
| 1. Tema 2: Automate finite:

2.2 verificarea acceptării unei secvențe implementare, testare și predare | Discutarea datelor de test, evaluare |  |
| 1. Tema 2 Automate finite

2.3 Rescrieți/adaptați programul de analiză lexicală (tema 1) astfel încât să folosească automate finite pentru a determina secvențele corespunzătoare atomilor lexicali  | Explicația, dialogul, studiu de caz, discutarea datelor de test, evaluare |  |
| 1. Tema 3: Implementarea unui analizor sintactic

3.1 alegerea structurilor de date și arhitectura aplicației | Explicația, dialogul, studiu de caz |  |
| 1. Tema 3: Implementarea unui analizor sintactic

3.2 implementarea funcțiilor principale ale analizorului | Explicația, dialogul, studiu de caz |  |
| 1. Tema 3: Implementarea unui analizor sintactic

3.3 programul principal și integrarea modulelor | Explicația, dialogul, studiu de caz |  |
| 1. Tema 3: Implementarea unui analizor sintactic

3.4 testare și corectarea posibilelor erori | Explicația, dialogul, studiu de caz |  |
| 1. Tema 3: Implementarea unui analizor sintactic

3.5 predarea  | Discutarea datelor de test, evaluare |  |
| 1. Tema 4: utilizare lex/flex + yacc/bison

4.1 implementare | Explicația, dialogul, studiu de caz |  |
| 1. Tema 4: utilizare lex/flex + yacc/bison

4.2 testare si predare | Discutarea datelor de test, evaluare |  |
| **Bibliografie** [1] K.D. COOPER, L. TORCZON - Engineering a Compiler,Elsevier Science & Technology, 2011.[2] Material on-line, zB. http://dinosaur.compilertools.net/ |

**9. Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului**

|  |
| --- |
| * Tematica cursului respectă recomandările de conținut IEEE si ACM pentru studiile din domeniul informatică
* Cursul există în programul de studiu al universităților importante din România și din alte țări
* Tematica cursului este considerată de companiile soft ca fiind importantă cel puțin pentru programatorii de nivel mediu
 |

**10. Evaluare**

|  |  |  |  |
| --- | --- | --- | --- |
| Tip activitate | 10.1 Criterii de evaluare | 10.2 metode de evaluare | 10.3 Pondere din nota finală |
| 10.4 Curs | - cunoasterea princiilor de baza ale domeniului- aplicarea conceptelor studiate la curs- rezolvarea de probleme | Examen scris | 75% |
| 10.5 Seminar/laborator | - sa fie capabili sa implementeze conceptele si algoritmii cursului | Verificare continua a activitatii in timpul orelor de laborator- verificarea documentatiei- verificarea programelor | 25% |
| 10.6 Standard minim de performanţă |
| Cel puțin nota 5 la examenul scris, și cel puțin nota 5 pentru media calculată. |

Data completării Semnătura titularului de curs Semnătura titularului de seminar

12.04.2024 Prof. dr. Klaus Dohmen Prof. dr. Klaus Dohmen

Data avizării în departament Semnătura directorului de departament

 Conf. dr. Adrian Sterca