

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babes-Bolyai Cluj-Napoca
1.2 Facultatea	Facultatea de Matematica si Informatica
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Informatica
1.5 Ciclul de studii	Licență
1.6 Programul de studiu / Calificarea	Informatică engleză

2. Date despre disciplină

2.1 Denumirea disciplinei	Generarea automată a programelor din algoritmi						
2.2 Titularul activităților de curs	Lect.univ.dr. Cioban Vasile						
2.3 Titularul activităților de seminar	Lect.univ.dr. Cioban Vasile						
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	Opțional

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

3.1 Număr de ore pe săptămână	3	Din care: 3.2 curs	2	3.3 seminar/laborator	1
3.4 Total ore din planul de învățământ	36	Din care: 3.5 curs	24	3.6 seminar/laborator	12
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					19
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					20
Tutoriat					10
Examinări					20
Alte activități:					
3.7 Total ore studiu individual		89			
3.8 Total ore pe semestru		125			
3.9 Numărul de credite		5			

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	•
4.2 de competențe	•

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	• Proiector
5.2 De desfășurare a seminarului/laboratorului	• lab cu stații cu o platformă de proiectare și programare

6. Competențele specifice acumulate

Competențe profesionale	- Produsele CASE asigura o gama variata de activitati, printre care: - scrierea automata a programelor pe baza specificatiilor prin Generatoare de cod; - redocumentarea, restructurarea, redezvoltarea sistemelor deja implementate prin programe de întreținere si dezvoltare. Aceasta acopera toate fazele din dezvoltarea programelor (analiza, proiectare, implementare, testare si distributie). - Clasic, un limbaj de programare este definit ca un limbaj artificial creat de om care serveste la exprimarea, sub forma de instructiuni executabile de catre calculator, a algoritmului de rezolvare a unei probleme. - Automatizarea translatării unui algoritm dintr-o formă de reprezentare de tip pseudocod într-un limbaj de programare.
Competențe transversale	- Învățarea unor scheme de prefixate de translatare automată. - Translatarea cu ajutorul operatorului uman. - Pe baza standardelor existente, de exemplu, compania ORACLE si-a definit propria metodologie CASE.

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	- Însușirea unor scheme de trasnlatare universale - Automatizarea translatării unui algoritm dintr-o formă de reprezentare de tip pseudocod într-un limbaj de programare.
7.2 Obiectivele specifice	- Învățarea unor scheme de prefixate de translatare automată. - Translatarea cu ajutorul operatorului uman.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Limbaje de programare (1 curs, Săpt.1) - definiție, clasificare, mod de specificare	Expunerea, modelarea, studiul de caz	
2. Translatare și translatoare (1 curs, Săpt.2) - definiție, rol, clasificare (interpretoare, compilatoare, link-editoare, etc)	Expunerea, modelarea, studiul de caz	
3. Automate finite și automate push-down. (1 curs, Săpt.3) - definiție, clasificare, utilizare în procesul de translatare;	Expunerea, modelarea, studiul de caz	
4. Instrumente CASE (2 cursuri, Săpt.4,5) - ciclul de viață al unui PP, etape care pot fi parțial automatizate;	Expunerea, modelarea, studiul de caz	
5. Instrumente CASE (2 cursuri, Săpt. 4,5) - definiție, realizare și utilizare;	Expunerea, modelarea, studiul de caz	
6. Generator de cod (1 curs, Săpt.6) - instrument CASE specific, în faza de codificare;	Expunerea, modelarea, studiul de caz	

7. Algoritmi și modalități de reprezentare (2 cursuri, Săpt. 7,8) - gramatici și automate speciale de reprezentare a structurilor de control și algoritmilor	Expunerea, modelarea, studiul de caz	
8. Algoritmi și modalități de reprezentare (2 cursuri, Săpt. 7,8) - gramatici și automate speciale de reprezentare a structurilor de control și algoritmilor	Expunerea, modelarea, studiul de caz	
9. Automatizarea translatarei algorimilor în cod de programare (2 cursuri, Săpt. 9,10) -algoritmi specifici.	Expunerea, modelarea, studiul de caz	
10. Automatizarea translatarei algorimilor în cod de programare (2 cursuri, Săpt. 9,10) -algoritmi specifici.	Expunerea, modelarea, studiul de caz	
11. Produse software care automatizează parțial translatarea. (2 cursuri, Săpt. 11,12) - produse soft specifice, exemple.	Expunerea, modelarea, studiul de caz	
12. Produse software care automatizează parțial translatarea. (2 cursuri, Săpt. 11,12) - produse soft specifice, exemple.	Expunerea, modelarea, studiul de caz	

Bibliografie

1. Luca Dan Serbanati –Limbe de programare si compilatoare, Ed. Academiei, 1987
2. Bazil Parv, Alexandru Vancea –Fundamentele limbajelor de programare, Ed. Alabastra, 1996
3. Ioan Salomie, Tehnici de programare obiectuală, ed. Alabastră, 1996
4. Bjarne Stroustrup, The C++ Programming Language, 3rd ed. Addison-Wesley, 1997
5. Bruce Eckel, Thinking in C++, Prentice Hall 2000
6. Stephen R. Schach, Software Engineering, edițiile 5, 6, 7, Ed Prentice Hall, 2002, 2005, 2007
7. www.cs.ubbcluj.ro/~vcioban/Informatica/Anul3/GenPro

8.2 Seminar / laborator	Metode de predare	Observații
1.Limbe de programare (Lab. 1, Săpt.1)	Expunerea, modelarea, exemplificarea	
2. Translatore și translatoare(Lab. 2, Săpt.3, tema 1)	Expunerea, modelarea, exemplificarea	
3. Automate finite și automate push-down. (Lab. 3, Săpt.5) - verificare Tema 1	Expunerea, modelarea, exemplificarea	
4. Generator de cod (Lab. 4, Săpt.7, Proiect)	Expunerea, modelarea, exemplificarea	
5. Algoritmi și modalități de reprezentare (Lab. 5, Săpt.9)	Expunerea, modelarea, exemplificarea	
6. Produse software care automatizează parțial translatarea. (Lab. 6, săpt.11)	Expunerea, modelarea, exemplificarea	

Bibliografie

- 1.Luca Dan Serbanati –Limbe de programare si compilatoare, Ed. Academiei, 1987
2. Bazil Parv, Alexandru Vancea –Fundamentele limbajelor de programare, Ed. Alabastra, 1996
3. Ioan Salomie, Tehnici de programare obiectuală, ed. Alabastră, 1996
4. Bjarne Stroustrup, The C++ Programming Language, 3rd ed. Addison-Wesley, 1997
5. Bruce Eckel, Thinking in C++, Prentice Hall 2000
6. Stephen R. Schach, Software Engineering, edițiile 5, 6, 7, Ed Prentice Hall, 2002, 2005, 2007
7. www.cs.ubbcluj.ro/~vcioban/Informatica/Anul3/GenPro

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

- Cursul este opțional și reprezintă o specializarea și aprofundare a domeniului translatării în cod sursă a algoritmilor;
- Utilizarea instrumentelor CASE poate automatiza procesul de codificare.

10. Evaluare

10. Evaluare			
Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	- să știe să aplice conceptele cursului	Colocviu (susținerea proiectului)	50%
	- rezolvarea de probleme		
10.5 Seminar/laborator	-să cunoscă exemplele prezentate în curs	Media temelor	50%
	-să rezolve problemele date ca tema		
10.6 Standard minim de performanță			
• pentru nota finală, studentul trebuie să ia nota 5 atât la colocviu cât și la fiecare temă de laborator (apoi se face media la laborator)			

Data completării

1.05.2016

Semnătura titularului de curs

lect.univ.dr. Vasile Cioban

Semnătura titularului de seminar

lect.univ.dr. Vasile Cioban

Data avizării în departament

.....

Semnătura directorului de departament

.....