

Fișa disciplinei

1. Date despre program

1.1. Instituția de învățământ	ACADEMIA DE STUDII ECONOMICE
1.2. Facultatea	CIBERNETICĂ, STATISTICĂ ȘI INFORMATICĂ ECONOMICĂ
1.3. Departamente	(Departament) INFORMATICĂ ȘI CIBERNETICĂ ECONOMICĂ
1.4. Domeniul de studii	Cibernetică, statistică și informatică economică
1.5. Ciclul de studii	LICENȚĂ
1.6. Forma de învățământ	CU FRECVENȚĂ
1.7. Programul de studii	Informatică economică
1.8. Limba de studiu	Română
1.9. Anul universitar	2018-2019

2. Date despre disciplina

2.1. Denumire	Structuri de date									
2.2. Cod	18.0205IF2.2-0004									
2.3. Anul de studii	2	2.4. Semestrul	2	2.5. Forma de evaluare	Examen	2.6. Regimul disciplinei	O (Obligato riu)	2.7. Nr. credite ECTS	4	
2.8. Titulari	C(C)	conf.univ.dr. POPA Marius Emanuel					marius.popa@ie.ase.ro			
	S(S)	conf.univ.dr. POPA Marius Emanuel					marius.popa@ie.ase.ro			
	S(S)	conf.univ.dr. DOINEA Laurențiu-Mihai					mihai.doinea@ie.ase.ro			
	S(S)	conf.univ.dr. CIUREA Cristian-Eugen					cristian.ciurea@ie.ase.ro			
	S(S)	lect.univ.dr. ZAMFIROIU I IONUȚ ALIN					alin.zamfiroiu@csie.ase.ro			

3. Timp total estimat

3.1. Număr de săptămâni	14.00
3.2. Număr de ore pe săptămână	4.00 din care
	C(C) 2.00
	S(S) 2.00
3.3. Total ore din planul de învățământ	56.00 din care
	C(C) 28.00
	S(S) 28.00
3.4. Total ore de studiu pe semestru (număr ECTS*25 ore)	100.00
3.5. Total ore studiu individual	44.00

<i>Distribuția fondului de timp pentru studiu individual</i>	
Studiu după manual, suport de curs, bibliografie și notițe	24.00
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	5.00
Pregătire seminarii, laboratoare, teme, referate, portofolii și eseuri	9.00
Tutoriat	1.00
Examinări	3.00
Alte activități	2.00

4. Precondiții

4.1. de curriculum	Bazele tehnologiei informației; Bazele programării calculatoarelor; Tehnici de programare; Programare orientată obiect
4.2. de competențe	Cunoașterea modului de organizare a datelor în memoria calculatorului, modele și cerințe de utilizare a datelor în programul sursă, utilizarea și alocarea zonelor de memorie; Cunoașterea elementelor de sintaxă a limbajului de programare C++ și a modului de construire, compilare, execuție și depanare a programului sursă.

5. Condiții desfășurare activități

pentru C(C)	Prelegeri desfășurate în săli cu echipament de predare multimedia (videoproiector)
pentru S(S)	Săli cu calculatoare instalate cu mediu de programare C++.

6. Competențe specifice acumulate

PROFESIONALE	C4	Dezvoltarea de componente pentru produse software, folosind structuri de date, algoritmi, tehnici și limbaje de programare evaluate
--------------	----	---

7. Obiectivele disciplinei

7.1. Obiectivul general	Inițiere în definirea și utilizarea adecvată a structurilor de date în dezvoltarea de componente pentru produse software.
7.2. Obiective specifice	Asimilarea tehnicilor și metodelor de proiectare și implementare a structurilor de date în procesul de dezvoltare a componentelor pentru produse software; Selectarea structurilor de date adecvate fiecărei componente software pe baza criteriilor de ordin practic; Utilizarea structurilor de date în proiecte software complexe orientate pe probleme economice.

8. Conținuturi

8.1. C(C)		Metode de predare/ lucru	Recomandări
1	Recapitulare. Tipuri de date standard și definite de programator. Pointeri. Modele și cerințe de definire, inițializare, utilizare și lizibilitate a datelor în programul sursă. Indicatori de performanță ai utilizării memoriei. Memoria stack și memoria heap.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare

2	Structuri de date dinamice necontigue: lista simplu înlănțuită și lista dublu înlănțuită – definire, alocare și utilizare. Stiva și coada – definire, alocare și utilizare.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
3	Matrice rare: definire, alocare și utilizare. Structuri de date neomogene și contigue. Implementarea matricelor rare prin masive și structuri de date neomogene.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
4	Structura de date de tip graf: caracteristici, definire, alocare și utilizare. Algoritmi de traversare a unui graf. Conectivitate și algoritmi de conectivitate.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
5	Structuri de date arborescente: arbore oarecare și arbore binar – definire, alocare și utilizare. Arbori de structură: caracteristici, implementare și operații.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
6	Structuri de date arborescente: arbore binar de căutare – definire, alocare și utilizare.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
7	Structuri arborescente echilibrate: arbori binari echilibrați, arbori AVL, arbori Roșu și Negru – definire operație, caracteristici ale arborilor echilibrați.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
8	Structura de date Arbore B: definire, proprietăți, alocare, algoritmi și implementarea operațiilor de bază (inserare și ștergere).	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
9	Tabele de dispersie: caracteristici, funcții hash, operații, mecanisme de evitare a coliziunilor.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
10	Structura Heap – definire, alocare și utilizare. Cozi de prioritate.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare
11	Compactarea și compresia datelor: caracteristici, clasificare și algoritmi de compresie a datelor. Conversii ale structurilor de date: caracteristici ale procesului de conversie, tipuri de conversie, elemente de eficiență la nivel de proces, respectiv la nivel de aplicație.	Suport PowerPoint, exemple de implementare în mediul de dezvoltare, interacțiune cu studenții	Distribuirea suportului de curs în format electronic, recomandarea de surse de documentare, exerciții de implementare

Bibliografie

- Ion Ivan, Marius Popa, Paul Pocatilu (coordonatori) , Structuri de date, ASE, București, 2008, România
- Ion Smeureanu , Programarea in limbajul C/C++, CISON, București, 2001, România
- <http://www.acs.ase.ro>, România
- <http://www.dice.ase.ro>, România
- Saumeyendra Sengupta, Carl Phillip Korobkin , C++ Object Oriented Data Structures, Springer Verlag, New York, 1994, Statele Unite ale Americii
- William Ford, William Topp, Data Structures with C++, Prentice Hall , New Jersey, 1996, Statele Unite ale Americii
- E. Demaine , Advanced Data Structures, 2003, http://courses.csail.mit.edu/6.897/spring03/scribe_notes, Statele Unite ale Americii

8.2. S(S)		Metode de predare/ lucru	Recomandări
1	Exemple de definire, inițializare și referire a datelor standard și a celor definite de programator. Accesul și referirea datelor prin pointeri. Utilizarea memoriilor stack și heap și legăturile dintre acestea. Determinarea indicatorilor de performanță în alocarea și utilizarea zonelor de memorie.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
2	Exemple pe liste simple și duble cu date standard și definite de utilizator. Implementarea și utilizarea listelor complet complet alocate în memoria dinamică. Utilizarea instrumentelor de depanare ale mediului de dezvoltare pentru asimilarea managementului zonelor de memorie alocate de programator. Exemple practice pentru implementarea și utilizarea stivelor și a cozilor utilizând structuri de date dinamice necontigue. Aplicație: implementarea formei postfixate a unei expresii matematice.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
3	Exemple practice pentru implementarea matricelor rare și a operațiilor pe și/sau cu matrice rare.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
4	Exemple de implementare a structurii de tip graf prin matrice de adiacență și listă de adiacență. Implementarea operațiilor de traversare unui graf.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
5	Exemple de implementare a algoritmilor de conectivitate.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
6	Exemple practice de implementare a structurilor arborescente. Implementarea operațiilor specifice de lucru cu structuri arborescente.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
7	Exemple practice de implementare a structurilor arborescente. Implementarea operațiilor specifice de lucru cu structuri arborescente.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
8	Exemple practice de implementare a operațiilor de echilibrare pentru arbori binari și arbori AVL.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
9	Implementarea și operații pe arbori de structură.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
10	Exemple practice de implementare a structurii Arbore B, implementarea algoritmilor pentru inserarea, respectiv ștergerea unei valori de cheie în structura Arbore B.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare
11	Implementarea tabelor de dispersie și a structurii Heap. Implementarea cozilor de prioritate.	Exemple de implementare în mediul de dezvoltare	Utilizarea calculatorului instalat cu mediul de programare

Bibliografie

- Ion Ivan, Marius Popa, Paul Pocatilu (coordonatori), Structuri de date, ASE, București, 2008, România
- Ion Ivan, Cristian Ionita, Cătălin Boja, Marius Popa, Adrian Pocovnicu, Daniel Milodin , Practica dezvoltării software orientată pe structuri de date, ASE, București, 2005, România
- Ion Smeureanu , Programarea in limbajul C/C++, CISON, București, 2001, România
- <http://www.acs.ase.ro>
- <http://www.dice.ase.ro>
- Bjarne Stroustrup , The C++ Programming Language, Addison-Wesley, <http://www.research.att.com/~bs/3rd.html> , Statele Unite ale Americii

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

În industria software, produsele obținute prezintă un nivel de complexitate deosebit de ridicat determinat de numărul de componente, legăturile dintre acestea, algoritmi implementați, funcționalități implementate și natura problemei căreia produsul software se adresează. În procesul de inginerie software, cunoașterea structurilor de date, a implementării acestora și a claselor de operații care sunt implementate pe baza lor este esențială în ameliorarea complexității produsului software și a procesului de dezvoltare a acestuia. Aceste elemente reprezintă factori esențiali în realizarea de produse software cu nivel calitativ ridicat, realizate cu costuri minime și în termenele planificate de către producătorii de software.

10. Evaluare

Activitatea	Criterii de evaluare	Metode de evaluare	Pondere în nota finală
10.1. S(S)	Examinare orală și/sau scrisă prin test la calculator, Evaluare orală și/sau scrisă a cunostintelor	Evaluarea structurilor de date implementate și răspunsurilor oferite	40.00
10.2. Evaluare finală	Examinare orală/scrise prin test la calculator, evaluarea orală și/sau scrisă a cunostintelor	Evaluarea structurilor de date implementate și răspunsurilor oferite	60.00
10.3. Modalitatea de notare	Note întregi 1-10		
10.4. Standard minim de performanță	<p>Grila de evaluare a cunostintelor:</p> <p>Nota 5: Gestiunea adreselor de memorie stack și heap, lucrul cu pointeri. Compilare, execuție și depanare aplicație în mediul de dezvoltare. Respectarea logicii de aplicație conform cerințelor de examinare.</p> <p>Nota 6: Definierea și alocarea adecvată în zone de memorie a structurilor de date solicitate a fi implementate în cadrul aplicației.</p> <p>Nota 7: Utilizarea adecvată în memorie a structurilor de date: inițializare și acces la structurile de date. Dezalocarea zonelor de memorie utilizate pentru structurile de date.</p> <p>Nota 8: Implementarea operațiilor solicitate pe structurile de date alocate conform cerințelor de examinare. Modularizarea aplicației.</p> <p>Nota 9: Definierea și gestiunea seturilor de date de test, validarea și gestiunea rezultatelor conform logicii de aplicație implementată.</p> <p>Nota 10: Elemente de optimizare a utilizării structurilor de date și management a zonelor de memorie.</p> <p>Observatii: Acordarea notelor mai mari sau egale cu 5 este conditionată de acumularea a minim 50% din punctele de evaluare acordate la evaluarea finală (prin examen).</p>		

Data listării,
17/01/2020

Titulari,

Data avizării în departament,

Director departament,