


UNIVERSITATEA TEHNICĂ
 DIN CLUJ-NAPOCA

FISA DISCIPLINEI
1. Date despre program

1.1	Institutia de invatamint superior	Universitatea Tehnica din Cluj-Napoca
1.2	Facultatea	Electronica, Telecomunicatii si Tehnologia Informatiei
1.3	Departamentul	Comunicatii
1.4	Domeniul de studii	Inginerie electronica si telecomunicatii
1.5	Ciclul de studii	Licența
1.6	Programul de studii/Calificarea	Tehnologii si Sisteme de Telecomunicatii/ Inginer
1.7	Forma de invatamint	IF - Invatamant cu frecventa
1.8	Codul disciplinei	EL3112

2. Date despre disciplina

2.1	Denumirea disciplinei		Programarea calculatoarelor-Algoritmi								
2.2	Aria tematica (subject area)		Inginerie electronica si telecomunicatii								
2.3	Responsabil de curs		Prof.dr.ing. Petre G. Pop								
2.4	Titularul disciplinei		Prof.dr.ing. Petre G. Pop								
2.5	Anul de studii	I	2.6	Semestrul	2	2.7	Evaluarea	Verificare pe parcurs	2.8	Regimul disciplinei	O

3. Timpul total estimat

An/ Sem	Denumirea disciplinei Ingineria programarii in electronica	Nr. sapt.	Curs			Aplicații			Stud. Ind.	TOTAL	Credit		
			[ore/săpt.]			[ore/sem.]							
			S	L	P	S	L	P					
III/Y		14	2	0	2	0	28	0	28	0	74	130	5

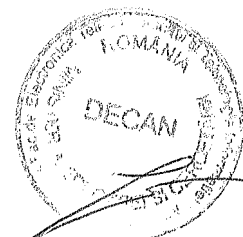
3.1	Numar de ore pe saptamina	2	3.2	din care curs	2	3.3	aplicatii	2
3.4	Total ore din planul de inv.	28	3.5	din care curs	28	3.6	aplicatii	28
Studiul individual								Ore
Studiul dupa manual, suport de curs, bibliografie si notite								30
Documentarea suplimentara in biblioteca, pe platformele electronice si pe teren								12
Pregatire seminarii/laboratoare, teme, referate, portofolii, eseuri								12
Tutoriat								2
Examinari								17
Alte activitati								1
3.7	Total ore studiul individual	74						
3.8	Total ore pe semestru	130						
3.9	Numar de credite	5						

4. Preconditii (acolo unde este cazul)

4.1	De curriculum	NU
4.2	De competente	NU

5. Conditii (acolo unde este cazul)

5.1	De desfasurare a cursului	NU
5.2	De desfasurare a aplicatiilor	Universitatea Tehnica din Cluj-Napoca



6 Competențe specifice acumulate

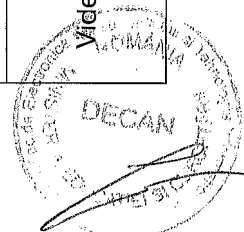
Competențe profesionale	Cunoștințe teoretice, (Ce trebuie să cunoască)	Concepte despre recursivitate, metode de programare recursive. Tehnici de cautare și sortare. Elemente de bază și avansate legate de programarea obiectuală: clase și obiecte, constructori, destructori, supraincarcarea funcțiilor și operatorilor, mostenirea simplă și multiplă, excepții, clase și funcții generice, clase și metode virtuale, sistemul de intrări/ieșiri obiectual. Structuri de date: stiva, coada, liste simplu înlanțuite, liste dublu înlanțuite, arbori, arbori binari.
	Deprinderi dobândite: (Ce știe să facă)	Dupa parcurgerea disciplinei studentii vor fi capabili: <ul style="list-style-type: none"> • Sa scrie aplicatii in limbajul C/C++: <ul style="list-style-type: none"> ○ ce implementeaza algoritmi recursivi; ○ ce efectueaza cautari si sortari; • Sa scrie aplicatii in limbajul C/C++: <ul style="list-style-type: none"> ○ ce folosesc clase si obiecte; ○ ce fac supraincarcari de functii si operatori; ○ ce folosesc mostenirea, metode virtuale; ○ ce citesc/scriu date la consola sau in fisiere text sau binare, folosind sistemul de intrari/iesiri obiectual; • Sa scrie aplicatii in limbajul C/C++: <ul style="list-style-type: none"> ○ ce folosesc structuri de date de tip stiva, coada, arbori;
	Abilități dobândite: (Ce instrumente știe să manuiască)	Dupa parcurgerea disciplinei studentii vor fi capabili: <ul style="list-style-type: none"> – Sa cunoasca principalele facilitati ale unui mediu de programare C/C++ cu facilitati obiectuale; – Sa depaneze si sa testeze aplicatii obiectuale si cu algoritmi complecsi in limbajul C/C++.
Competențe transversale	<ul style="list-style-type: none"> – Identificarea unor obiective specifice de realizat, a resurselor disponibile, a condițiilor de implementare și a etapelor, termenelor, timpilor aferente realizării unei aplicații software simple; – Utilizarea eficientă a surselor informaționale și a resurselor de comunicare și formare profesională asistată de calculator (internet, aplicații software de specialitate, baze de date, cursuri și seminarii online etc.) atât în limba română, cât și într-o limbă de circulație internațională. 	

7 Obiectivele disciplinei (reiesind din grila competențelor specific acumulate)

7.1	Obiectivul general al disciplinei	Dezvoltarea de competențe profesionale în domeniul proiectării, realizării și testării aplicațiilor software
7.2	Obiectivele specifice	Asimilarea cunoștințelor teoretice și practice privind proiectarea și realizarea aplicațiilor software ce folosesc algoritmi, structuri de date și programarea obiectuală.

8. Continuturi

8.1. Curs (programa analitică)		Metode de predare	Observații
1	Recursivitate: funcții recursive, ieșirea din recursivitate, încărcarea stivei. Metode de programare recursive: metoda divide et impera.	Expunere, discuții	Video-proiector
2	Metode de programare recursive: backtracking. Tehnici de cautare. Tehnici simple de sortare: prin selecție, inserție, interschimbare. Tehnici avansate de sortare: merge-sort, quick-sort.		
3	Programare obiectuală (POO): problematica POO. Clase, obiecte, membri, autoreferință.		



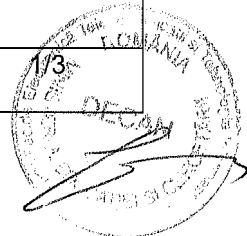
4	Constructori. Destructori. Constructor de copiere, obiecte ca date membre, tablouri de obiecte, domenii de vizibilitate.		
5	Functii si clase prietene. Membri statici. Structuri si reuniuni in C++.		
6	Supraincercarea operatorilor.		
7	Conceptul de mostenire, mostenirea simpla si multipla.		
8	Clase virtuale, metode virtuale, clase abstracte.		
9	Sistemul de i/e in C++: biblioteca iostream, i/e cu format, starea sistemului de i/e, functii manipulator.		
10	Clasele ostream si istream, redefinirea operatorilor de i/e, fisiere C++.		
11	Stive. Cozi. Liste. Liste statice.		
12	Liste dinamice: LSI, LDI.		
13	Arbori. Definitie, proprietati. Arbori binari, definitie, operatii.		
14	Comunicare note. Situatiia finala.		
8.2. Aplicatii (lucrari)		Metode de predare	Observatii
1	Aplicatii bazate functii macro, functii inline, functii cu numar variabil de parametri, functii cu parametri impliciti	Expunere si aplicatii	Calculatorul, softuri de simulare avansata, montaje experimentale de laborator, echipamente specifice pentru masurare
2	Functii recursive. Metode de programare recursive: divide et impera.		
3	Metoda de programare backtracking. Tehnici de cautare.		
4	Tehnici de sortare.		
5	Clase, obiecte, membri, accesul la membri.		
6	Accesul la membrii unei clase folosind modificatori de acces.		
7	Constructori. Destructori. Tablouri de obiecte.		
8	Functii si clase prietene. Membri statici.		
9	Clase si functii virtuale. Clase abstracte.		
10	Supraincercarea operatorilor.		
11	Mostenirea simpla si multipla.		
12	Intrari/iesiri C++. Redefinirea operatorilor de i/e.		
13	Fisiere in C++.		
14	Evaluare finala.		
<p>In biblioteca UTC-N</p> <ol style="list-style-type: none"> Liviu Negrescu, Limbajul C si C++ pentru incepatori, vol 1-4, Ed. Albastra, 2005, 2006. Ligia Chiorean, Mircea - F. Vaida, Petre G. Pop, Cosmin Striletschi, Elemente de baza si obiectuale privind dezvoltarea aplicatiilor in limbajul de programare C/C++, Ed. UTPress, 2008. <p>In alte biblioteci</p> <ol style="list-style-type: none"> M. F. Vaida, P.G. Pop, C. Striletschi, Ligia D. Chiorean, Lenuta Alboaie, Programarea in limbajul C/C++. Algoritmi de baza in C/C++, Ed. Casa Cartii de Stiinta, Cluj-Napoca, 2011. M. F. Vaida, Lenuta Alboaie, P.G. Pop, Ligia D. Chiorean, C. Striletschi, Programare orientata pe obiecte si programare web in C/C++, Ed. Casa Cartii de Stiinta, Cluj-Napoca, 2011. <p>Materiale didactice virtuale</p> <p>Materiale suport de curs si laborator disponibile pe site-ul laboratorului : http://helios.utcluj.ro/lab.</p>			

9. Coroborarea continuturilor disciplinei cu asteptarile reprezentantilor comunitatii epistemice, asociatiilor, profesionale si angajatori din domeniul aferent programului

Competentele achizitionate vor fi necesare angajatilor in urmatoarele ocupatii posibile conform COR (Inginer electronist, transporturi, telecomunicatii; Proiectant inginer electronist; Proiectant inginer de sisteme si calculatoare; Inginer proiectant comunicatii) sau in noi ocupatii propuse pentru a fi incluse in COR (Inginer suport vanzari; Dezvoltator de aplicatii multimedia; Inginer operare retea; Inginer testare sisteme de comunicatii; Manager proiect; Inginer de trafic; Consultant pentru sisteme comunicatii)

10. Evaluare

Tip activitate	10.1	Criterii de evaluare	10.2	Metode de evaluare	10.3	Ponderea din nota finala
Activitate laborator si teme de casa		Activitatea saptamanala la laborator. Teme de casa : 10 seturi de probleme; (nota L)		Verificare pe parcurs		1/3
Curs		Evaluarea cunostintelor teoretice (nota T)		Test teoretic		1/3



Aplicatii		Evaluarea cunostintelor practice (nota P)		O problema rezolvata printr-un program		1/3
10.4 Standard minim de performanta						
Formula de calcul a notei: $N = (L+T+P)/3$; se calculeaza daca: $L > 4$ si $T > 4$ si $P > 4$						

Data completarii 24.07.2012
Titularul de disciplina Prof.dr.ing Petre G. Pop

Responsabil de curs Prof.dr.ing. Petre G. Pop

Data avizarii in departament 01.10.2012

Director departament Prof.dr.ing. Virgil Dobrota

