

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Facultatea de Electronică, Telecomunicații și Tehnologia Informației
1.3 Departamentul	Comunicații
1.4 Domeniul de studii	Inginerie electronică, telecomunicații și tehnologii informaționale
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Tehnologii și Sisteme de Telecomunicații/ Inginer
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	TST11.00

2. Date despre disciplină

2.1 Denumirea disciplinei	Programarea calculatoarelor și limbaje de programare II						
2.2 Aria de conținut	Arie teoretică Arie metodologică Arie de analiză						
2.3 Responsabil de curs	Prof.dr.ing. Gavril Petre POP, petre.pop@com.utcluj.ro						
2.4 Titularul activităților de seminar / laborator / proiect	Prof.dr.ing. Gavril Petre POP, petre.pop@com.utcluj.ro As.drd.ing. Mihai Eusebiu JECAN, eusebiu.jecan@com.utcluj.ro						
2.5 Anul de studiu	I	2.6 Semestrul	2	2.7 Tipul de evaluare	Verificare	2.8 Regimul disciplinei	DF/DI

3. Timpul total estimat

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar / laborator	2
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar / laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					34
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					9
Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri					20
Tutoriat					2
Examinări					4
Alte activități:					
3.7 Total ore studiu individual			69		
3.8 Total ore pe semestru			125		
3.9 Numărul de credite			5		

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Cunoștințe însușite la cursurile anterioare de programarea calculatoarelor
4.2 de competențe	NU

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Amfiteatru cu videoproiector și tablă, Cluj-Napoca
5.2. de desfășurare a laboratorului	Sala de laborator, stații de lucru, un mediu de programare instalat pe fiecare stație de lucru, Cluj-Napoca

6. Competențele specifice acumulate

Competențe profesionale	C3. Aplicarea cunoștințelor, conceptelor și metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, microcontrolere, limbaje și tehnici de programare C4. Conceperea, implementarea și operarea serviciilor de date, voce, video, multimedia, bazate pe înțelegerea și aplicarea noțiunilor fundamentale din domeniul comunicațiilor și transmisiunii informației C5. Selectarea, instalarea, configurarea și exploatarea echipamentelor de telecomunicații fixe sau mobile și echiparea unui amplasament cu rețele uzuale de telecomunicații
Competențe transversale	N / A

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1 Obiectivul general al disciplinei	Dezvoltarea de competențe privind algoritmi de calcul, structuri de date și programare obiectuală.
7.2 Obiectivele specifice	Asimilarea cunoștințelor teoretice și practice de bază privind proiectarea, implementarea și testarea unor aplicații în tehnologie obiectuală, folosind limbajul C++.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Elemente de programare C++ non-OOP. Recursivitate: funcții recursive, ieșirea din recursivitate, încărcarea stivei. Exemple.	Expunere la tablă, prezentare cu videoproiector, discuții.	
2. Metode de programare recursive și nerecursive. Backtracking. Divide et impera. Tehnici de căutare.		
3. Tehnici simple de sortare. Tehnici avansate de sortare.		
4. Introducere în programarea obiectuală (POO): problematica POO. Clase, obiecte, membri, autoreferință. Constructori. Destructorii.		
5. Constructor de copiere, obiecte ca date membre, tablouri de obiecte, domenii de vizibilitate. Funcții și clase prietene. Membri statici. Structuri și reuniuni în C++.		
6. Supraîncărcarea metodelor. Supraîncărcarea operatorilor (considerații generale, variante de supraîncărcare, operatori de conversie). Supraîncărcarea operatorilor (=, ++, --).		

7. Supraîncărcarea operatorilor (new, delete, [], ()). Conceptul de moștenire. Moștenirea simplă. Moștenirea multiplă.		
8. Constructori și destructori pentru clasa derivată. Clase virtuale, metode virtuale, clase abstracte.		
9. Destructori virtuali. Excepții în C++. Sistemul de I/E în C++: biblioteca iostream, I/E cu format, starea sistemului de I/E.		
10. Funcții manipulator. Clasele ostream și istream, redefinirea operatorilor de i/e, fișiere C++.		
11. Structuri de date abstracte. Tipuri: set, map, deque. Structuri de date: stive, cozi, liste, arbori.		
12. Funcții și clase generice. Biblioteca STL.		
13. Biblioteca STL (cont.).		
14. Evaluare teorie		
Bibliografie <ol style="list-style-type: none"> Mircea-Florin Vaida, Petre G. Pop, Cosmin Strilețchi, Ligia Chiorean, Calin G. Login, Tehnologii avansate privind dezvoltarea aplicațiilor software în limbajul C/C++, Casa Cartii de Stiinta, 2006 Ligia Chiorean, Mircea-Florin Vaida, Petre G. Pop, Cosmin Strilețchi, , Elemente de baza si obiectuale privind dezvoltarea aplicațiilor în limbajul de programare C/C++, UTPress, 2007/2008 Mircea-Florin Vaida, Ligia-Domnica Chiorean, Lenuta Alboai, Petre Gavril Pop, Cosmin Strilețchi, Kuderna-Iulian Benta, Programarea în limbajul C/C++ cu elemente C++1y. Programare web C++, Casa Cartii de Stiinta, Cluj-Napoca, 2016 Materiale suport de laborator disponibile pe site-ul colectivului IT: https://helios.utcluj.ro/learn2code 		
8.2 Seminar / laborator / proiect	Metode de predare	Observații
1. Aplicații bazate funcții macro, funcții inline, funcții cu număr variabil de parametri, funcții cu parametri implicați	Prezentarea obiectivelor, a fundamentelor teoretice. Exemplificări practice. Lucrul individual și în echipă. Evaluare.	
2. Funcții recursive		
3. Metode de programare recursive și nerecursive. Backtracking, divide et impera. Tehnici de căutare.		
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. Funcții și clase prietene. Membri statici.		
9. Supraîncărcarea operatorilor.		
10. Moștenirea simplă și multiplă.		
11. Clase și funcții virtuale. Clase abstracte.		
12. Intrări/ieșiri C++. Redefinirea operatorilor de i/e.		
13. Fișiere în C++. Evaluarea temelor.		
14. Evaluare finală.		
Bibliografie <ol style="list-style-type: none"> Ligia-Domnica Chiorean, Kuderna-Iulian Benta, Mircea-Florin Vaida, Petre Gavril Pop, Cosmin Strilețchi, C/C++ - Ghid teoretic si practic, Casa Cartii de Stiinta, Cluj-Napoca, 2016. Materiale suport de laborator disponibile pe site-ul colectivului IT: https://helios.utcluj.ro/learn2code 		

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

Competențele dobândite vor fi folosite în următoarele ocupații conform COR (Clasificarea Ocupațiilor din România): Inginer emisie; Inginer electronist, transporturi, telecomunicații; Inginer imagine; Inginer sunet; Proiectant inginer electronist; Proiectant inginer de sisteme și calculatoare; Inginer șef car reportaj; Inginer șef schimb emisie; Inginer proiectant comunicații; Inginer sisteme de securitate; Inginer suport vânzări; Dezvoltator de aplicații multimedia; Inginer operare rețea; Inginer testare sisteme de comunicații; Manager proiect; Inginer de trafic; Consultant pentru sisteme de comunicații.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
Aplicații (lucrări)	Evaluarea activității săptămânale (nota L)	Verificare pe parcurs	20%
Aplicații (evaluare)	Evaluare practică (nota P)	Problema de rezolvat pe calculator	40%
Curs	Test teoretic (nota T)	Test grilă	40%

10.6 Standard minim de performanță

Nivel calitativ:

Cunoștințe minimale:

- Elemente teoretice de bază privind recursivitatea, căutarea și sortarea
- Conceptele de bază ale programării obiectuale

Competențe minimale:

- Să scrie, implementeze, testeze programe de complexitate medie, ce folosesc căutări și sortări
- Să scrie, implementeze, testeze programe de complexitate medie ce folosesc clase și obiecte

Nivel cantitativ:

- Efectuarea tuturor lucrărilor de laborator
- $T > 4$ și $P \geq 4.50$
- Nota la disciplină se calculează cu formula: $(0.2 * L + 0.4 * P + 0.4 * T)$

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
20.06.2023	Curs	Prof.dr.ing. Gavril Petre POP	
	Aplicații	Prof.dr.ing. Gavril Petre POP	
		As.drd.ing. Mihai Eusebiu JECAN	

Data avizării în Consiliul Departamentului COM 11.07.2023	Director Departament Comunicații Prof.dr.ing. Virgil DOBROTA
Data aprobării în Consiliul Facultății ETTI 12.07.2023	Prof.dr.ing. Ovidiu POP