

FIȘA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea „Tibiscus” din Timișoara
1.2. Facultatea	Calculatoare și Informatică Aplicată
1.3. Departamentul	Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii/Calificarea	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	Programare 1 – LIN114						
2.2. Titularul activității de curs	Lect.univ.dr. Simona Angela APOSTOL						
2.3. Titularul activității de laborator	Inf.Ing. Tiberiu-Gabriel DRAGOTĂ						
2.4. Anul de studiu	1	2.5. Semestrul	1	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	Conținut=DF Obligatorietate=DO

3. Timpul total estimat

3.1. Numărul de ore pe săptămână	4	din care 3.2. curs	2	3.3. laborator	2
3.4. Total ore din planul de învățământ	56	din care 3.5. curs	28	3.6. laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					28
Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren					22
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					26
Tutoriat					14
Examinări					4
Alte activități					-
3.7. Total ore studiu individual					59
3.8. Total ore pe semestru					125
3.9. Numărul de credite					5

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Algebră și analiză matematică nivel liceu, algoritmi elementari, scheme logice
4.2. de competențe	Gândire algoritmică, capacitatea de a analiza o problemă și de a o descompune în pași logici; Utilizarea calculatorului, a mediilor de dezvoltare simple și a instrumentelor de scriere a codului; Capacitatea de a asimila concepte noi și de a aplica cunoștințele în contexte practice.

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Cursul se desfășoară în sala dotată cu videoproiector, tablă interactivă sau whiteboard, permițând expunerea interactivă a conținutului teoretic și exemplificarea în timp real a conceptelor de programare.
5.2. de desfășurare a seminarului/laboratorului	Calculatoare sau stații de lucru cu sistem de operare actualizat; Medii de programare integrate (IDE) adecvate limbajului C/C++(ex: Code::Blocks, Visual Studio); Conexiune la internet pentru accesarea resurselor online, a platformei educaționale (Google Classroom) și pentru transmiterea temelor.

6. Obiectivele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei

Cunoștințe	- explică principiile programării procedurale, orientate obiect, logice și funcționale; - descrie structura și funcționarea unui program informatic în limbaje moderne (C/C++); - recunoaște eficiența și complexitatea algoritmilor, optimizând performanța codului;
Abilități	- scrie, testează, documentează și optimizează codul în limbaje de programare moderne (C/C++) - utilizează paradigme de programare diferite (procedurală, orientată obiect, funcțională, logică); - identifică și rezolvă probleme de programare prin metode sistematice de testare și depanare;
Responsabilitate și autonomie	- coordonează echipe pentru dezvoltarea de aplicații informatice, asumând decizii responsabile legate de optimizarea și integrarea acestora. - produce software și îl adaptează continuu la noile tehnologii și cerințe de piață.

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

7.1. Obiectivele generale ale disciplinei	Studentii să înțeleagă și să se familiarizeze cu standardul ANSI C și să poată scrie programe în limbajul C. Elaborarea și analiza unor algoritmi pentru rezolvarea problemelor: însușirea noțiunilor elementare de algoritmică, implementarea algoritmilor în limbajul C.
7.2. Obiectivele specifice	La finalizarea cu succes a acestei discipline, studenții vor fi capabili să: Identifice și utilizeze principiile programării procedurale. Lucreze într-un limbaj procedural în scopul înțelegerii și asimilării cunoștințelor necesare unui bun programator. Sa se familiarizeze cu limbajul de programare cel mai utilizat Prezinte elementele de bază ale limbajului și dobândirea unei minime experiențe în utilizarea acestora. Scrie în pseudo-cod, scheme logice și limbaj de programare algoritmi cu diferite grade de complexitate. Realizeze aplicații în limbajul C, pentru testarea funcționării instrucțiunilor și operatorilor, funcțiilor și procedurilor recursive; fișierelor; pointerilor în C.

8. Conținuturi

8.1. Curs	Metode de predare	Observații
1. Generalități algoritmi, limbaj pseudocod Introducere în limbajul C; Elementele de bază ale limbajului	Expunerea interactivă, documentarea pe web, problematizarea, conversația euristică, exemplificarea.	Calculator Videoproiector Tabla Smart
Limbajul C 2.1. Structura programelor C. Sintaxa programelor C. 2.2. Tipuri simple de date. 2.3.Instrucțiuni de citire/scriere, atribuire, de selecție, repetitive,etc. 2.4. Tipuri de date definite de utilizator. 2.5. Subprograme. Proceduri. Funcții 2.7. Fișiere binare, text, prelucrari de fișiere 2.8. Tablouri 2.10.Structuri 2.11.Tehnici de programare - Divide et Impera, Greedy, Backtracking		Calculator Videoproiector Tabla Smart
Elemente de baza ale limbajului C++ Tipuri de date derivate și utilizator și alocare dinamică în C++ Metoda programarii orientate-obiect in C++		
Bibliografie: ● S.Apostol – Programare Procedurală, Tehnici Avansate de programare – note de curs. ● B. Kernigham si D. Ritchie – The C Programming Language, 2nd ed., ● Negrescu Liviu – Limbajele C Si C++ Pentru Incepatori (Vol.1) PDF (scribd.com) ● P.J.Plauger - The Standard C Library, Prentice Hall, 2022. ● Herbert Schildt - "C: Manual de referinta". Ed.Teora. 2020		

8.2. Seminar/laborator	Metode de seminarizare	Observații
1. Introducere în programare. Prezentare mediului de lucru Code Blocks	Operare la calculator. Familiarizarea cu mediul de lucru. Operare la calculator. Realizarea problemelor propuse în lucrarea de laborator	Studenții vor primi materialul pe Classroom
2. Primele programe în C		
3. Funcții ,operatori, instrucțiuni		
4. Funcții recursive		
5. Tablouri unidimensionale, multidimensionale		
6. Pointeri		
7. Structuri		
8. Tehnici de programare-D&I, Greedy		
9. Metodele Backtracking		
10. Limbajul C++	Elaborarea, tastarea și rularea programelor exemplificative Elaborarea, tastarea și rularea programelor exemplificative	
11. Alocare dinamică în C++		
12. Programare orientată obiect în C++		
13.-14. Test practic		

Bibliografie:

Referințe principale:

Simona Apostol – Programare Procedurală (-îndrumător de laborator-), Editura Victor Babeș, Timișoara, 2021.

Jeff Szuhay - Learn C Programming: A beginner's guide to learning C programming the easy and disciplined way, Editura Packt, June 26 2020

www.timsoft.ro/aux/module - teorie + teste+ probleme rezolvate

Referințe suplimentare:

Liviu Negrescu – Limbajele C și C++ pentru începători, vol 1., partea 1, Ed. Albastru, 2020.

H. Schildt – C – manual complet, Ed. Teora, actualizată 2021.

Dr. K. Jamsa, L. Klander – Totul despre C și C++ - manualul fundamental de programare în C și C++, Ed. Teora, actualizată 2011.

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

Cursul și laboratorul vor furniza studenților abilități și cunoștințe elementare de programare în limbajul C/C++, atât la nivelul necesar unui profesor pentru predarea matematicii în liceu, cât și la nivelul de start în urmarea unei filiere de studiu în informatică.

Conținutul este în concordanță cu structura cursurilor similare de la alte universități și acoperă aspectele fundamentale necesare familiarizării cu problematica programării.

Abilitatea de a identifica, proiecta, implementa și analiza probleme care se pot rezolva este esențială pentru orice activitate din domeniul informaticii.

10. Evaluare

Tip de activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.1. Curs	Evaluarea are în vedere următoarele categorii de cunoștințe: Cunoștințe generale și de detaliu	Examinare scrisă test cuprinzând 4 modalități de testare a cunoștințelor: • Expunere a unor subiecte desemnate. • Întrebări cu variante multiple de răspuns orientate spre noțiunile cheie. • Întrebări grilă cu variante multiple de răspuns cu secvențe de cod din programe. Rezolvarea unor probleme propuse.	60 %
10.2. Seminar/laborator	• Cunoștințe generale: utilizarea limbajului de programare 'C' de bază. • Cunoștințe de detaliu: utilizarea algoritmilor și realizarea de programe simple de utilizare a limbajului C • Cunoștințe avansate: rezolvarea de probleme complexe, de dificultate sporită.	- Evaluarea activității la laborator - Participarea activă la activitățile de laborator - Rezolvarea problemelor propuse ca teme la laboratoare. Examinare printr-un test practic pe calculator cuprinzând probleme specifice limbajului cu diferite grade de dificultate.	40 %

10.3. Standard minim de performanță

Examinare scrisă:

- Pentru nota 5 este necesară tratarea minimală a fiecăreia din cele 4 categorii de subiecte.
- Studentul trebuie să poată defini noțiunile de bază cerute; să recunoască și să facă diferențe între expresiile condiționale, operanzi vs operatori; să știe să scrie în C bibliotecile, sintaxa programului principal, calculul sumelor și produselor, calcul matricial și polinomial, utilizarea stringurilor, lucrul cu fișiere.
- Pentru nota 10 este necesară tratarea tuturor subiectelor menționate pentru nota 5 însă în toată complexitatea lor.

Probe practice și activitate de laborator:

- Pentru nota 5 studentul trebuie să poată edita, compila, depana și rula programe C în mod consolă, programe de complexitatea unui începător în programare: calculul sumelor și produselor, calcul matricial și polinomial, utilizarea stringurilor, lucrul cu fișiere. (minim rezolvarea unei teme date la laborator și rezolvarea unei probleme din cele 3 propuse pe biletul de examen).
- Pentru nota 10 este necesară obținerea unui nivel superior (min. 80%) pentru cunoștințe avansate prezentate anterior respectiv rezolvarea celor 3 probleme propuse pe biletul de examen în toată complexitatea lor și problemele trebuie să fie executate/compilate cu succes.

Data completării

20.09.2025

Titular de disciplină

Lect.univ.dr. Simona Apostol

Data avizării în departament

26.09.2025

Director de departament

Conf.univ.dr. Victoria IORDAN