

FIȘA DISCIPLINEI¹⁾

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiesti
1.2. Facultatea	Litere si Stiințe
1.3. Departamentul	Informatică, Tehnologia Informației, Matematică, Fizică
1.4. Domeniul de studii universitare	Informatică
1.5. Ciclul de studii universitare	Licență
1.6. Programul de studii universitare	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	Fundamentele Programarii
2.2. Titularul activităților de curs	Lector dr. Tudorică Daniela
2.3. Titularul activităților seminar/laborator	Lector dr. Dobre Iuliana
2.4. Titularul activității proiect	-
2.5. Anul de studiu	I
2.6. Semestrul*	1
2.7. Tipul de evaluare	Examen
2.8. Categoria formativă** / regimul*** disciplinei	DF/O

*numărul semestrului este conform planului de învățământ;

**DF - Discipline fundamentale; DD - discipline de domeniu; DS - discipline de specialitate; DC - discipline complementare, DA - disciplina de aprofundare, DSI- disciplina de sinteza.

***obligatorie = O; opțională = A; facultativă = L

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

3.1. Număr de ore pe săptămână	4	din care: 3.2. curs	2	3.3. Seminar/laborator	2	3.4. Proiect	-
3.5. Total ore din planul de învățământ	56	din care: 3.6. curs	28	3.7. Seminar/laborator	28	3.8. Proiect	-
3.9. Distribuția fondului de timp							ore
Studiu după manual, suport de curs, bibliografie și notițe							29
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren							20
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri							28
Tutoriat							14
Examinări							3
Alte activități							-
3.10 Total ore studiu individual	94						
3.11. Total ore pe semestru	150						
3.12. Numărul de credite	6						

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	
4.2. de competențe	-

¹⁾ Adaptare după Ordinul Ministrului educației, cercetării, tineretului și sportului nr. 5 703/2011 privind implementarea Codului național al calificărilor din învățământul superior, publicat în Monitorul Oficial al României, partea I, nr.880 bis / 13.XII.2011

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	➤ Sală cu dotare clasică și mijloace de proiecție
5.2. de desfășurare a seminarului/laboratorului	➤ Sală dotată cu calculatoare și software adecvat

6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none">➤ C1.1 Descrierea adecvată a paradigmelor de programare și a mecanismelor de limbaj specificeC1.3 Elaborarea codurilor sursă adecvate și testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificațiiC1.4 Testarea unor aplicații pe baza unor planuri de test.C4.1 Definirea conceptelor și principiilor de bază ale informaticii, precum și a teoriilor și modelelor matematice
Competențe transversale	CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională

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

7.1. Obiectivul general al disciplinei	➤ Obiectivul principal al disciplinei constă în însușirea cunoștințelor de bază necesare programării într-un limbaj de programare procedural, de către studenții anului I
7.2. Obiectivele specifice	După parcurgerea disciplinei studenții vor putea să: <ul style="list-style-type: none">➤ definească și să identifice structurile algoritmice fundamentale necesare rezolvării unei probleme cu ajutorul calculatorului➤ descrie în pseudocod și să codifice în limbajul de programare C algoritmul de rezolvare pentru problemele propuse în cadrul laboratorului➤ aplice cunoștințele acumulate pentru rezolvarea informatică și a altor probleme, cu un grad de dificultate asemănător cu cele abordate la laborator

8. Conținuturi

8.1. Curs	Nr.ore	Metode de predare	Observații
1.Concepte de bază. Dată, informație, cunoștințe. Paradigme de programare.	4	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	

Conceptul de programare structurată Algoritmi-caracteristici, descriere. Teorema de structură. Reprezentarea structurilor algoritmice fundamentale. Tipuri de date si operații elementare. Structuri de date. Proceduri si functii (pseudocod și schemă logică) Translatarea, linkeditarea și încărcarea programelor pentru execuție.			
2.Structura generală a unui program C. Directive preprocesor Elemente de bază ale limbajului C.Tipuri de date fundamentale în C, variabile, constante. Functii uzuale de intrare-iesire pentru consolă. Descriptori de format. Operatori în C. Clase de precedență. Conversii de tip explicite si implicite	4	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
3.Instrucțiuni de control ale programului (instrucțiuni expresie, instrucțiuni de decizie, instrucțiuni iterative, instrucțiuni de salt).	2	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
4.Tablouri,inițializarea tablourilor, siruri de caractere . Funcții uzuale pentru prelucrarea șirurilor de caractere.	2	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
5..Pointeri-declarație și inițializare. Pointeri și tablouri unidimensionale. Pointeri către tipul char și șiruri de caractere. Pointeri către tablouri unidimensionale. Operații aritmetice cu pointeri. Pointeri dubli și tablouri bidimensionale.	2	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
6. Structuri. Uniuni. Campuri de biti, enumerări, declarații typedef	4	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
7. Funcții utilizator- declarație și definiție. Transferul datelor către funcții (prin valoare și prin referință). Argumentele funcției main(). Pointeri la funcții. Funcții recursive.	4	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
8. Clase de memorare	2	Expunere pe baza suportului	

		de curs, studii de caz, conversație, dezbatere	
9. Funcții pentru prelucrarea fișierelor (Funcții pentru deschidere și închidere fișier-fopen() și fclose(). Funcții pentru citirea și scrierea unui caracter-fgetc() și fputc(). Funcții pentru transferul sirurilor de caractere-fgets() și fputs(). Funcții pentru transfer cu format-fscanf() și fprintf(). Funcții pentru transferul blocurilor de date-fread() și fwrite(). Funcțiile feof() și rewind(). Funcții pentru citirea și modificarea indicatorului de poziție al fișierului -fseek(), fgetpos(), fsetpos(), ftell()).	4	Expunere pe baza suportului de curs, studii de caz, conversație, dezbatere	
Bibliografie <ol style="list-style-type: none"> 1. Brookshear J. Glenn, Introducere în Informatică, Editura Teora, București, 1999 2. Cristian Marinoiu, Programarea în limbajul C, Editura Universității din Ploiești, Ploiești, 2000 3. Cristian Marinoiu, Elena Simona Nicoară, Introducere în Informatică, Editura Universității din Ploiești, Ploiești, 2014 4. Brian Kernighan, Dennis Ritchie, The C programming Language, Prentice Hall, 1988, disponibilă la http://www.ime.usp.br/~pf/Kernighan-Ritchie/C-Programming-Ebook.pdf 5. Herbert Schildt C: Manual complet, Editura Teora, București, 1997 			
8.2. Seminar / laborator	Nr. ore	Metode de predare	Observații
1.Exemple de algoritmi; reprezentarea lor în pseudocod și schemă logică	2	Expunere, studii de caz, conversație, dezbatere: aplicații pe calculator	
2.Programe în limbajul C care utilizează instrucțiuni de intrare/ieșire, instrucțiuni de decizie și instrucțiuni iterative și tablouri de date	10	Expunere, studii de caz, conversație, dezbatere: aplicații pe calculator	
3. Programe care utilizează structuri. uniuni.campuri de biti, enumerări, declarații typedef și fișiere	16	Expunere, studii de caz, conversație, dezbatere: aplicații pe calculator	
Bibliografie <ol style="list-style-type: none"> 1. Brookshear J. Glenn, Introducere în Informatică, Editura Teora, București, 1999 2. Cristian Marinoiu, Programarea în limbajul C, Editura Universității din Ploiești, Ploiești, 2000 3. Cristian Marinoiu, Elena Simona Nicoară, Introducere în Informatică, Editura Universității din Ploiești, Ploiești, 2014 4. Brian Kernighan, Dennis Ritchie, The C programming Language, Prentice Hall, 1988, disponibilă la http://www.ime.usp.br/~pf/Kernighan-Ritchie/C-Programming-Ebook.pdf 5. Herbert Schildt C: Manual complet, Editura Teora, București, 1997 			

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

- Cursul și laboratorul sunt astfel concepute încât, prin competențele formate, să răspundă cerințelor pieței muncii.
- Disciplina Programarea procedurală există în planul de învățământ al specializărilor de Informatică din universitățile românești

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	Calitatea prezentării și a răspunsurilor la examinarea finală	Examinare orală cu bilete	25%
10.5. Seminar/laborator	Calitatea activității desfășurate în cadrul laboratorului	Intrebari privitoare la modul de rezolvare a problemelor propuse in cadrul laboratorului	25%
	Calitatea răspunsurilor la testele de control	Teste de control pe parcursul semestrului	25%
	Calitatea rezolvării pe calculator a aplicației la examinarea finală	Examinare orală cu bilete: aplicație pe calculator	25%
10.6. Standard minim de performanță			
<ul style="list-style-type: none"> ➤ Să descrie corect și să exemplifice structurile algoritmice fundamentale în limbajul C, să poată elabora și implementa algoritmi care necesită structuri de decizie și de iterație, să utilizeze funcții pentru implementarea unui program în C 			

Data completării Semnătura titularului de curs Semnătura titularului de seminar/laborator Semnătura titularului de proiect

9.09.2020 Lector dr. Tudorică Daniela Lector Dr. Dobre Iuliana _____

Semnătura directorului de departament

Data avizării în departament

Conf. dr. Gabriela Moise
