

**FIȘA DISCIPLINEI**  
Anul universitar 2015-2016

Decan,  
Prof. Corneliu Lazăr

**1. Date despre program**

1.1 Instituția de învățământ superior	Universitatea Tehnică „Gheorghe Asachi” din Iași
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și tehnologia informației
1.5 Ciclul de studii <sup>1</sup>	Licență
1.6 Programul de studii	Tehnologia informației

**2. Date despre disciplină**

2.1 Denumirea disciplinei	<b>Programare Orientată pe Obiecte</b>						
2.2 Titularul activităților de curs	Ș.l. dr.ing. Robert Gabriel Lupu						
2.3 Titularul activităților de aplicații	Asist. drd. inf. Ionuț Gavrilă						
2.4 Anul de studii <sup>2</sup>	2	2.5 Semestrul <sup>3</sup>	3	2.6 Tipul de evaluare <sup>4</sup>	Examen	2.7 Tipul disciplinei <sup>5</sup>	DID

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

3.1 Număr de ore pe săptămână	5	din care 3.2 curs	3	3.3a sem.	-	3.3b laborator	2	3.3c proiect	-
3.4 Total ore din planul de învățământ <sup>6</sup>	70	din care 3.5 curs	42	3.6a sem.	-	3.6b laborator	28	3.6c proiect	-
Distribuția fondului de timp <sup>7</sup>								Nr. de ore	
Studiul după manual, suport de curs, bibliografie și notițe								36	
Documentare suplimentară în bibliotecă sau pe platformele electronice de specialitate								14	
Pregătire laboratoare, teme de casă, test pe parcurs								20	
Tutoriat								7	
Examen final								3	
Alte activități:								-	
3.7 Total ore studiu individual <sup>8</sup>	80								
3.8 Total ore pe semestru <sup>9</sup>	150								
3.9 Numărul de credite	6								

**4. Precondiții (acolo unde este cazul)**

4.1 de curriculum <sup>10</sup>	<b>- Programarea Calculatoarelor</b>
4.2 de competențe	<b>- Programare structurată</b>

**5. Condiții (acolo unde este cazul)**

5.1 de desfășurare a cursului <sup>11</sup>	Tablă, videoproiector
5.2 de desfășurare a laboratorului <sup>12</sup>	Sală de laborator cu calculatoare și acces la internet Medii de programare: Microsoft Visual Studio, NetBeans

**6. Competențele specifice acumulate<sup>13</sup>**

Număr de credite alocate disciplinei <sup>14</sup> :			6	Repartizare credite pe competențe <sup>15</sup>
Competențe profesionale	CP1	Operarea cu fundamente științifice, ingineresti și ale informaticii	2	
	CP2	Proiectarea componentelor hardware, software și de comunicații	1,3	
	CP3	Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor	1	
	CP4	Proiectarea și integrarea sistemelor informatice utilizând tehnologii și medii de programare	1,4	
	CP5	Întreținerea și exploatarea sistemelor hardware, software și de comunicații	-	
	CP6	Utilizarea sistemelor inteligente	-	
Competențe transversale	CT1	Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei	0,1	
	CT2	Identificarea, descrierea și derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă și descrierea clară și concisă, verbal și în scris, în limba română și într-o limbă de circulație internațională, a rezultatelor din domeniul de activitate	0,1	
	CT3	Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională	0,1	

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

7.1 Obiectivul general al disciplinei	Asimilarea principiilor programării orientate pe obiect.
7.2 Obiective specifice	Aplicarea principiilor POO in limbajele C++ și Java Familiarizare cu biblioteca STL (Standard Template Library) pentru C++ și Colecții Java

**8. Conținutul disciplinei**

8. 1 Curs	Metode de predare	Obs.
<p><b>1. Capitol introductiv (6 ore)</b></p> <p>1.1 Structura cursului, bibliografie, mod de evaluare, competențe dobândite</p> <p>1.2 Principalele tehnici de programare - monolitică, procedurală, modulară, orientată pe obiecte, funcțională</p> <p>1.3 Recapitulare C</p> <p>1.4 Facilități ale limbajului C++ - tipuri de date, spații de nume, domeniu, I/O stream, gestiunea memoriei, tipul referință, argumente cu valori predefinite, supraîncărcarea numelor de funcții, constante, pointeri și constante, pointeri pe void, struct layout, câmpuri de biți, forward declaration, funcții inline</p> <p><b>2. Principiile de bază ale POO (12 ore)</b></p> <p>2.1 Prezentare generală – abstractizare, moștenire, polimorfism</p> <p>2.2 Tipul abstract de dată – Clasa în C++ - abstractizarea și încapsularea datelor, specificatori de acces, constructori de initializare, listă de inițializare, destructor, constructor de copiere, funcții membre inline, membri statici, pointerul this, funcții membre constante, funcții și clase prietene, operatorii <i>new</i>, <i>delete</i>, <i>new []</i> și <i>delete []</i>, pointeri la membri, supraîncărcarea operatorilor</p> <p>2.3 Moștenire în C++ - moștenirea, ierarhii de tipuri, specificatori de acces, suprascrierea metodelor din clasa de bază, ordinea de apelare a constructorilor și a destructorilor, constructori apelați implicit, rescrierea funcțiilor din clasa de bază, moștenirea multiplă, moștenire virtuală, studiu de caz – problema diamantului</p> <p>2.4 Polimorfism în C++ - pointeri la clasa de bază, operatorul de referențiere &amp;, tip declarat / tip real, apelarea unei funcții suprascrise, funcții virtuale, tabela virtuală, apelare constructori și destructori în cazul polimorfismului, destructori virtuali, polimorfism – exemplu. Funcții virtuale pure, type casting - <i>dynamic_cast</i>, <i>static_cast</i>, <i>reinterpret_cast</i></p> <p><b>3. Programare generică în C++ (2 ore)</b></p> <p>3.1 Funcții template</p> <p>3.2 Clase template</p> <p><b>4. Standard Template Library C++ (6 ore)</b></p> <p>4.1 Prezentare generală</p> <p>4.2 Containeri – secvențiale, asociative, adaptive</p> <p>4.3 Iteratori</p> <p>4.4 Algoritmi</p> <p>4.5 Tratarea erorilor</p> <p><b>5. Clase, obiecte și relații (2 ore)</b></p> <p><b>Principiile POO în Java (14 ore)</b></p> <p><b>1. Capitol introductiv</b></p> <p>1.1 Structura cursului, bibliografie, medii de dezvoltare</p> <p>1.2 Prezentare generală a limbajului</p> <p>1.3 Comparații C++ &amp; Java – sintaxă, semantică, managementul resurselor</p> <p><b>2. Obiecte în Java</b></p> <p>2.1 Tipuri primitive de date</p>	<p>Expunere la tablă și și cu ajutorul video proiectorului</p>	<p>Actualizare periodică a conținutului cursului.</p> <p>Notele de curs sunt disponibile în format electronic</p>

2.2 Spatii de nume 2.3 Tipul referință 2.4 Pachete de clase 2.5 Metode statice și ale obiectelor 2.6 Vectori <b>3.Derivare, moștenire polimorfism în Java</b> <b>4.Clase abstracte si interfețe</b> <b>5.Tratarea erorilor</b> <b>6. Fire de execuție</b> <b>7. Colecții – liste, multimi, dicționare, iteratori, interfețe, set, map</b>		
<b>Bibliografie</b> 1) The C++ programming language fourth edition, Bjarne Stroustrup 2) Programming principles and practice using C++, Bjarne Stroustrup 3) Effective C++, third edition, Scott Meyers 4) Inside the C++ object model, Stanley B. Lippman 5) Bruce Eckel - www.BruceEckel.com 6) Thinking in C: Foundations for C++ and Java 7) Thinking in C++, 2nd edition 8) Tutorial - http://www.cplusplus.com/ 9) Curs practic de Java, Cristian Frăsinaru, Editura Matrixrom 10) Java de la 0 la expert, Stefan Tanasă, Editura Polirom 11) Head first Java, Kathy Seirra & Bert Bates, Editura O'REILLI 12) Thinking in Java, Bruce Eckel, editura Prentice Hall 13) Oracle, Java SE 7 Fundamentals 14) Android Apps Performance Optimization, Herve Guihot, Apress		
<b>8. 2 Laborator</b>	<b>Metode de predare</b>	<b>Obs.</b>
1. Introducere in MS Visual Studio, cin/cout, new/delete. 2. Namespace, domeniu de valabilitate și vizibilitate, funcții cu parametri implicați, Supraîncărcarea funcțiilor. 3. Constructori de inițializare, destructor. 4. Membri statici, constructor de copiere, this. 5. Pointeri la clase, lista de inițializare. 6. Funcții și clase prietene, supraîncărcarea operatorilor. 7. Moștenire. 8. Moștenire multiplă, moștenire virtuală. 9. Pointeri la clasa de bază, metode virtuale, polimorfism, case abstracte. 10. Funcții și clase template. 11. Introducere în Java, prezentarea mediului integrat NetBeans, crearea unui proiect, exerciții 12. Clasa, modificatori de acces, constructori, redefinirea metodelor 13. Variabile si metode statice, clase abstracte si interfețe 14. Realizarea unui proiect complex în Java	Partea teoretică și temele de laborator sunt puse la dispoziția studenților la începutul semestrului. În timpul laboratorului, se discută cerințele și soluțiile posibile pentru temele solicitate. Studenții rezolvă problemele individual sau în echipă, primesc asistență și sunt evaluați.	

**9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului<sup>16</sup>**

Disciplina se regăsește în planurile de învățământ a tuturor departamentelor de Calculatoare din țară și străinătate, conținutul disciplinei fiind similar cu cel unanim acceptat în comunitatea epistemică. Competențele, cunoștințele și abilitățile dobândite sunt esențiale pentru pregătirea oricărui inginer, sunt solicitate de toți angajatorii din domeniul IT și se regăsesc în testele și interviurile de la angajarea absolvenților.

**10. Evaluare**

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Cunoștințe teoretice însușite (cantitatea, corectitudinea, acuratețea)	Teste pe parcurs <sup>22</sup> : 2 teste, săptămânile 5 și 10	30 %
		Examinare finală: probă practică	60 % (minim 5)

10.5 Laborator	Finalizarea programelor și verificare lor, interpretarea rezultatelor, evaluarea preciziei simulării etc.	Evaluare continuă pe parcursul orelor de laborator	10 % (minim 5)
10.6 Standard minim de performanță			
Abilități de programare în C++ Abilități de programare în Java Selecția și utilizarea independentă a tehnicilor învățate pentru rezolvarea unor probleme tip cunoscute; Capacitatea de a dezvolta aplicații utilizând tehnica programării orientată pe obiecte centrate pe particularitățile limbajului C++ respectiv Java.			

Data completării,  
22 Sept. 2014

Titular de curs,  
Ș.I.dr.ing. Robert Gabriel Lupu

Titulari de laborator,  
Asist.drd.ing. Ionuț Gavrilă

Data avizării în departament:  
25 Sept. 2014

Director de departament,  
Prof. Petru Cașcaval

<sup>1</sup> Licență / Master

<sup>2</sup> 1-4 pentru licență, 1-2 pentru master

<sup>3</sup> 1-8 pentru licență, 1-3 pentru master

<sup>4</sup> Examen, colocviu sau VP A/R – din planul de învățământ

<sup>5</sup> DF - disciplină fundamentală, DID - disciplină în domeniu, DS – disciplină de specialitate sau DC - disciplină complementară - din planul de învățământ

<sup>6</sup> Este egal cu 14 săptămâni x numărul de ore de la punctul 3.1 (similar pentru 3.5, 3.6abc)

<sup>7</sup> Liniile de mai jos se referă la studiul individual; totalul se completează la punctul 3.7.

<sup>8</sup> Suma valorilor de pe liniile anterioare, care se referă la studiul individual.

<sup>9</sup> Suma dintre numărul de ore de activitate didactică directă (3.4) și numărul de ore de studiu individual (3.7); trebuie să fie egală cu numărul de credite alocate disciplinei (punctul 3.9) x 24 de ore pe credit.

<sup>10</sup> Se menționează disciplinele obligatorii a fi promovate anterior sau echivalente

<sup>11</sup> Tablă, videoproiector, flipchart, materiale didactice specifice etc.

<sup>12</sup> Tehnică de calcul, pachete software, standuri experimentale, etc.

<sup>13</sup> Competențele din Grilele G1 și G1bis ale programului de studii, adaptate la specificul disciplinei, pentru care se repartizează credite ([www.mccis.ro](http://www.mccis.ro) sau site-ul facultății)

<sup>14</sup> Din planul de învățământ

<sup>15</sup> Creditele alocate disciplinei se distribuie pe competențe profesionale și transversale în funcție de specificul disciplinei

<sup>16</sup> Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

<sup>17</sup> Creditele alocate disciplinei se distribuie pe competențe profesionale și transversale în funcție de specificul disciplinei

<sup>18</sup> Titluri de capitole și paragrafe

<sup>19</sup> Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții (pentru fiecare capitol, dacă este cazul)

<sup>20</sup> Demonstrație practică, exercițiu, experiment

<sup>21</sup> Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

<sup>22</sup> Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii.

<sup>23</sup> Se va preciza numărul de teste și săptămânile în care vor fi susținute.