

SYLLABUS

pentru disciplina:

“PROGRAMARE ORIENTATĂ PE OBIECTE”

FACULTATEA AUTOMATICĂ ȘI CALCULATOARE  
DOMENIUL /SPECIALIZAREA INFORMATICĂ

Anul de studii: II

Semestrul ( 1 sau 2) 1

<b>Titularul cursului: Vasile Stoicu-Tivadar</b> <b>Colaboratori: Dorin Berian</b>					
<b>Numar de ore/saptamana/Verificarea/Credite</b>					
<b>Curs</b>	<b>Seminar</b>	<b>Laborator</b>	<b>Proiect</b>	<b>Examinare</b>	<b>Credite</b>
<b>2,5</b>	<b>0</b>	<b>1,5</b>	<b>0,5</b>	<b>D</b>	<b>5</b>

**A. OBIECTIVELE CURSULUI**

Transmiterea cunoștințelor de bază despre paradigma programării orientate pe obiecte, crearea deprinderilor teoretice și practice necesare analizei, proiectării, codificării și testării aplicațiilor de complexitate medie în limbajul C++.

Înțelegerea modului în care cunoștințele despre programarea orientată pe obiecte sînt utilizabile în programarea Windows.

**B. SUBIECTELE CURSULUI**

Paradigma programării orientate pe obiecte. (2h)

Limbajul C++. Generalități (programarea în stilul C++, operatori de I/E, variabile locale, clase, supraîncărcare, moștenire, constructori/destructori). (2h)

Clase și obiecte (clase, structuri, uniuni, funcții prietene, funcții inline, constructori, membri statici, manipularea obiectelor). (4h)

Matrice, pointeri și referințe (matrice de obiecte, pointeri către obiecte, referințe, alocarea dinamică). (4h)

Supraîncărcarea (supraîncărcarea funcțiilor, supraîncărcarea operatorilor). (4h)

Moștenirea. Polimorfismul cu funcții virtuale (proprietăți, utilizare). (4h)

Operații de I/E în C++ (stream-uri, clase utilizate, variabila de stare, formatarea datelor, supraîncărcarea operatorilor << și >>, lucrul cu fișiere).(4h)

Studiu de caz: introducere în programarea Windows (stilul windows, funcții API, bucle de mesaje, ierarhia MFC). (2h)

Tipare. Containere, iteratori, algoritmi, colecții. Standard Template Library. (2h)

**C. SUBIECTELE APLICATIILOR (laborator, seminar, proiect)**

Laborator:

Completări aduse de C++ față de limbajul C. (2h)

Încapsularea prin intermediul claselor. (2h)

Pointeri la metode. (2h)

Constructori, destructori. (2h)

Funcții și clase friend. (2h)

Derivarea claselor. Moștenirea multiplă.(4h)

Metode virtuale. (3h)

Utilizarea listelor eterogene. (4h)

Proiect: Teme individuale de complexitate medie incluzând utilizarea tiparelor. (7h)

Ex. Aplicație de editare grafică prin crearea și utilizarea unei ierarhii de clase specifice.

**D. BIBLIOGRAFIE** *Se indică maximum trei titluri bibliografice de referință*

1. Stoicu-Tivadar, V., Programare orientată pe obiecte, Editura Orizonturi Universitare, Timisoara, 2000

2 Schildt, H., “C++ manual complet”, Editura Teora, București, 1997.

3. Somnea, D., Turturea, D., “Inițiere în C++. Programarea orientată pe obiecte”, Editura Tehnică, București, 1993

## **E. PROCEDURA DE EVALUARE**

*Nota de activitate pe parcurs este obținută prin ponderarea notelor de la laborator și proiectz conform timpului alocat. Notele vor ține cont de implicarea în activitate, de calitatea acesteia, de modul de prezentare.*

*Examinarea este distribuită în 2 probe: test-grilă (scris, 60 întrebări referitoare la conținutul cursului, din care cel puțin 20 cu conținut preponderent practic - cu cod sursă), proba pe calculator (problemă de C++, 1 h 30 min), nota de examen are pondere 2/3 din nota finală.*

## **F.COMPATIBILITATE INTERNATIONALA**

*Toate universitățile cu profil informatic sau știința calculatoarelor, exemple:*

Technische Universitat Wien, UMIST Manchester, California Institute of Technology, Universitatea Tehnică și Economică din Budapesta (Budapest University of Technology and Economics) etc.

Data: 10.04.2008

**DIRECTOR/SEF DEPARTAMENT/CATEDRA**

**Conf. dr. ing. Ioan Silea**

**TITULAR DE DISCIPLINĂ,**

**Prof. dr. ing. Vasile Stoicu-Tivadar**