

# Fișa disciplinei

## 1. Date despre program

1.1. Instituția de învățământ	ACADEMIA DE STUDII ECONOMICE
1.2. Facultatea	CIBERNETICĂ, STATISTICĂ ȘI INFORMATICĂ ECONOMICĂ
1.3. Departamente	(Departament) INFORMATICA SI CIBERNETICA ECONOMICA
1.4. Domeniul de studii	Cibernetică, statistică și informatică economică
1.5. Ciclul de studii	LICENȚĂ
1.6. Forma de învățământ	CU FRECVENȚĂ
1.7. Programul de studii	Informatică economică
1.8. Limba de studiu	Română
1.9. Anul universitar	2016-2017

## 2. Date despre disciplina

2.1. Denumire	<b>Programare Orientată Obiect</b>									
2.2. Cod	<b>16.0205IF2.1-0001</b>									
2.3. Anul de studii	<b>2</b>	2.4. Semestrul	<b>1</b>	2.5. Forma de evaluare	<b>Examen</b>	2.6. Regimul disciplinei	<b>O (Obligato riu)</b>	2.7. Nr. credite ECTS	<b>5</b>	
2.8. Titulari	C(C)	<b>prof.univ.dr. SMEUREANU Ion</b>					ion.smeureanu@csie.ase.ro			
	C(C)	<b>conf.univ.dr. BOJA Cătălin Emilian</b>					catalin.boja@ie.ase.ro			
	C(C)	<b>lect.univ.dr. VINTE Claudiu</b>					claudiu.vinte@ie.ase.ro			
	C(C)	<b>prof.univ.dr. POCATILU Paul</b>					paul.pocatilu@ie.ase.ro			
	S(S)	<b>conf.univ.dr. BOJA Cătălin Emilian</b>					catalin.boja@ie.ase.ro			
	S(S)	<b>prof.univ.dr. SMEUREANU Ion</b>					ion.smeureanu@csie.ase.ro			
	S(S)	<b>lect.univ.dr. VINTE Claudiu</b>					claudiu.vinte@ie.ase.ro			
	S(S)	<b>conf.univ.dr. DOINEA Laurențiu-Mihai</b>					mihai.doinea@ie.ase.ro			
	S(S)	<b>conf.univ.dr. CIUREA Cristian-Eugen</b>					cristian.ciurea@ie.ase.ro			
	S(S)	<b>asist.univ.dr. ZAMFIROIU I IONUȚ ALIN</b>					alin.zamfiroiu@csie.ase.ro			
	S(S)	<b>asist.univ.dr. IANCU M BOGDAN</b>					bogdan.iancu@ie.ase.ro			
	S(S)	<b>Asociat drd. ENĂCHESCU C MIHAELA IRINA</b>					irina.enachescu@csie.ase.ro			

## 3. Timp total estimat

3.1. Număr de săptămâni	14.00
3.2. Număr de ore pe săptămână	4.00 din care
	C(C) 2.00
	S(S) 2.00

3.3. Total ore din planul de învățământ	56.00	din care		
			C(C)	28.00
			S(S)	28.00
3.4. Total ore de studiu pe semestru (număr ECTS*25 ore)	125.00			
3.5. Total ore studiu individual	69.00			
<i>Distribuția fondului de timp pentru studiu individual</i>				
Studiu după manual, suport de curs, bibliografie și notițe	30.00			
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	15.00			
Pregătire seminarii, laboratoare, teme, referate, portofolii și eseuri	20.00			
Tutoriat				
Examinări	4.00			
Alte activități				

#### 4. Precondiții

4.1. de curriculum	Bazele Tehnologiei Informatiei, Bazele Programarii Calculatoarelor, Algoritmi si tehnici de programare
4.2. de competențe	-

#### 5. Condiții desfășurare activități

pentru C(C)	Prelegerile de curs se desfășoară în săli cu echipament de predare multimedia;
pentru S(S)	Seminariile se desfășoară în săli cu acces la Internet si PC. Mediul de dezvoltare utilizat este Microsoft Visual Studio 2010 sau 2012

#### 6. Competențe specifice acumulate

PROFESIONALE	C4	Dezvoltarea de componente pentru produse software, folosind structuri de date, algoritmi , tehnici si limbaje de programare evaluate
PROFESIONALE	C5	Dezvoltarea de aplicatii informatice care utilizeaza baze de date, resurse multimedia si tehnologii client-server/servicii web

#### 7. Obiectivele disciplinei

7.1. Obiectivul general	Asimilarea conceptelor de programare orientata obiect si dezvoltarea capacitatii studentilor de rezolvarea a problemelor practice prin elaborarea de programe dezvoltate in paradigma orientata obiect.
7.2. Obiective specifice	- Familiarize students with the concepts specific to object-oriented programming; - Help students to understand the C++ language syntax and logic elements behind object oriented program design.

#### 8. Conținuturi

8.1. C(C)		Metode de predare/ lucru	Recomandări
1	Recapitularea unor elemente ale programarii procedurale: functii, transferul parametrilor, pointeri la date si functii, clase de memorie.	Prelegere si interacțiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.

2	Conceptele de clasa, obiect, constructor, destructor, metode de acces, pointerul this. Declararea si implementarea metodelor in clasa si in afara clasei.	Prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
3	Asimilarea conceptelor de constructor de copiere, supraincarcare operator =, supraincarcare operatorilor unari si binari (+, -, ++, etc), obiecte cu extensii in memoria dinamica si domenii de nume (namespace).	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
4	Conversii intre diferite tipuri de obiecte (operatorul cast, operatorul= si constructor de copiere), vector de obiecte, modificatorul const, tipologia membrilor statici (static), obiecte constante, pointeri constanti la obiecte si pointeri la obiecte constante.	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
5	Mecanismul try-catch in C++.	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
6	Supraincarcarea operatorilor	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
7	Clase derivate, mostenire. Polimorfism.	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
8	Functii virtuale, supradefinire, mostenire multipla	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.

9	Mecanisme de tip RTTI, mostenire multipla si dynamic cast	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
10	Functii si clase template. Modalitati de incapsulare a functiilor si claselor sablon in biblioteci statice/dinamice	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
11	Operatii I/O orientate pe stream-uri. Implementarea obiectuala a structurilor dinamice de date (liste, arbori, etc.). Conceptele de serializare/deserializare obiecte.	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
12	Lucru cu biblioteci dinamice in C++.	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.
13	Standard Template Library - STL – containere, iteratori si algoritmi. Clasa string, map, list, vector, etc.	Prelegere bazata pe prezentari multimedia si interactiune cu studentii.	- Suportul de curs este postat pe platforma programului de licenta. - Se recomandă studentilor parcurgerea prealabilă a suportului de curs pentru a putea interacționa mai ușor cu profesorul în timpul predării.

### **Bibliografie**

- Ion Smeureanu, Marian Dardala , Programarea orientata obiect in limbajul C++, CISON, Bucuresti, 2002, România
- Ion Smeureanu, Programarea in limbajul C/C++, CISON, Bucuresti, 2001, România
- Bjarne Stroustrup , The C++ Programming Language, 3rd Edition, Addison-Wesley, <http://www.research.att.com/~bs/3rd.html>, Statele Unite ale Americii
- Herbert Schildt, C++ manual complet, Teora, Bucuresti, România

8.2. S(S)		Metode de predare/ lucru	Recomandări
1	Dezvoltarea și rularea programelor sub Microsoft Visual Studio .NET in C++. Exemplu de structura cu pointer la functie (pentru comportament - subliniata legatura intre structura si clasa)	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.

2	Exemplu practic pentru clasa Student reliefand conceptele de clasa, obiect, constructor, destructor, metode de acces, pointer this. Declararea metodelor in clasa si in afara clasei.	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.
3	Exemplu practic de clasa pentru asimilarea conceptelor de constructor de copiere, supraincarcare operator =, supraincarcare operatori unari si binari (+, -, ++, etc), obiecte cu extensii in memoria dinamica si domenii de nume (namespace).	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.
4	Exemple integrative pentru conceptele clasa, obiecte, supraincarcare operatori, transferul obiectelor in/din functii, metode friend, pointer this, etc. prin construirea claselor BigInt, Data, Vector si Matrice. Mostenire simpla, functii virtuale si polimorfism.	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.
5	Generalizarea claselor prin obtinerea de sabloane (template)	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.
6	Evaluare prin test practic	-	-
7	Implementarea conceptului de serializare	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.
8	Standard Template Library - STL – containere, ieratori si algoritmi. Clasa string, map, list, vector, etc. Exemple practice.	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.
9	Evaluare proiect	Sustinere individuala a proiectului. Incadrarea in cerintele de realizare si prezentare.	Incadrarea in cerintele de realizare si prezentare.
10	Clase derivate, mostenire si polimorfism utilizand pointerul la tabela pointerilor catre functii virtuale. Exemplu pe clasele Muncitor/Angajat/Profesor – polimorfism prin supraincarcare si polimorfism prin derivare/pointer la obiect/si functii virtuale.	Interactiune cu studentii in vederea dezvoltarii aplicatiei si analizei programului dezvoltat la seminar	Se recomandă studentilor parcurgerea suportului de curs si seminar/laborator din platforma a programului de licenta pentru a putea interacționa mai ușor cu profesorul.

11	Mecanismul try-catch.	Interacțiune cu studenții în vederea dezvoltării aplicației și analizei programului dezvoltat la seminar	Se recomandă studenților parcurgerea suportului de curs și seminar/laborator din platforma a programului de licență pentru a putea interacționa mai ușor cu profesorul.
12	Conversii între diferite tipuri de obiecte (operatorul cast, operator = și constructor de copiere), vector de obiecte, modificatorul const, tipologia membrilor statici (static), obiecte constante, pointeri constanti la obiecte și pointeri la obiecte constante.	Interacțiune cu studenții în vederea dezvoltării aplicației și analizei programului dezvoltat la seminar	Se recomandă studenților parcurgerea suportului de curs și seminar/laborator din platforma a programului de licență pentru a putea interacționa mai ușor cu profesorul.

### **Bibliografie**

- Ion Smeureanu, Marian Dardala, Programarea orientată obiect în limbajul C++, CISON, București, 2002, România
- Ion Smeureanu, Programarea în limbajul C/C++, CISON, București, 2001, România
- Bjarne Stroustrup, The C++ Programming Language, 3rd Edition, Addison-Wesley, <http://www.research.att.com/~bs/3rd.html>, Statele Unite ale Americii
- Herbert Schildt, C++ manual complet, Teora, București, România

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

Conținutul disciplinei este discutat cu specialiști și cu personalități din universități și companii de marcă ce activează în domeniul IT&C. Cursul asigură un nivel de bază pentru orice viitor programator.

## **10. Evaluare**

Activitatea	Criterii de evaluare	Metode de evaluare	Pondere în nota finală
10.1. S(S)	Evaluare practică - test la calculator	Evaluări pe parcurs prin teste practice. Proiect.	20.00
10.2. S(S)	Evaluare practică la laborator - proiect	Evaluări pe parcurs prin teste practice. Proiect.	10.00
10.3. S(S)	Evaluare participare laborator - teme	Evaluări pe parcurs prin teme sau prin participare activă la seminar	10.00
10.4. Evaluare finală	Evaluarea cunoștințelor acumulate	Test grila și dezvoltarea unei aplicații C++ pe baza unor specificații descrise în biletul de examen. Aplicația se dezvoltă utilizând Visual Studio în timpul alocat examenului. Evaluarea subiectului este orală.	60.00
10.5. Modalitatea de notare	Note întregi 1-10		
10.6. Standard minim de performanță	Compilare și rulare în linia de comandă/mediu și depanare de aplicații C++. Modelarea unor aspecte din realitate folosind principiile POO și implementare software. Implementarea unei soluții fără erori. Obținerea a minim 50% din punctajul aferent evaluărilor din timpul semestrului Realizarea proiectului		

Data listării,  
05/10/2016

Titulari,

Data avizării în departament,

Director departament,