

## Limbajul de programare C++

**Durata:** 20 – 40 ore

**Infrastructura folosită:** retroproiector, tablă, calculator personal pe care este instalat un mediu de dezvoltare C++ (editor, compilator, editor de legături, depanator, biblioteci standard, inclusiv STL). De preferat este un mediu integrat de dezvoltare C++, de exemplu Microsoft Visual C++ Express Edition.

**Cine poate participa:** sunt două categorii de participanți

- programatori care vin din programarea procedurală (de exemplu C) și doresc să intre în lumea programării orientate pe obiecte prin intermediul lui C++
- programatori care folosesc deja alte limbaje orientate pe obiecte (de exemplu Java) și care doresc să se familiarizeze și cu C++.

**Cerințe pentru participanți:** cunoașterea limbajului de programare C facilitează parcurgerea elementelor comune C – C++.

**Efectul cursului:** participanții vor beneficia de o introducere solidă în limbajul de programare C++, în noțiunile de bază ale programării orientate pe obiecte și suportul oferit de limbaj în această direcție.

**Cursuri conexe:** Analiza și proiectarea orientată pe obiecte, Șabloane de proiectare

**Verificarea cunoștințelor:** opțional, pe parcurs și/sau test final

**Bibliografie:** The C++ Programming Language, Third Edition, Bjarne Stroustrup, Addison-Wesley, ISBN 0-201-88954-4

**Descriere:** Cursul se adresează în principal dezvoltatorilor de software ce au deja experiență cu programarea în C și care doresc să facă trecerea la programarea orientată pe obiecte prin intermediul lui C++.

Cursul arată locul programării orientate pe obiecte între paradigmele de programare, introduce noțiunea de clasă sau tip definit de utilizator, relații între clase, polimorfism, programare generică, tratarea erorilor, supraîncărcarea operatorilor, biblioteca standard C++ – Standard Template Library (STL).

Se vor evidenția principiile programării orientate pe obiecte: abstractizarea, ascunderea informației, moștenirea, agregarea, încapsularea, polimorfism, programare generică.

Noțiunile teoretice se fixează prin teme și probleme practice care de obicei au o pondere de cel puțin 50% din durata cursului.

## Conținut:

1. Introducere în programarea orientată pe obiecte (POO) și C++
2. Paradigme de programare: procedurale, modulare, abstractizarea datelor, programare orientată pe obiecte
3. Diferențe față de C; parcurgerea elementelor comune
4. Suport pentru abstractizare datelor: inițializarea și distrugerea obiectelor, asignarea și inițializarea, șabloane, tratarea excepțiilor, conversii de tip
5. Suport pentru POO: clase abstracte și clase concrete, implementări multiple, implementarea virtualizării funcțiilor, moștenirea multiplă
6. Obiecte și clase: definirea unei clase, folosirea unei clase, accesul la membrii unei clase, domeniul de vizibilitate clasă, definirea unei clase într-o altă clasă, declararea incompletă a unei clase, folosirea membrilor dată a unei clase, date statice, obiecte ca și date membru, pointeri la membrii dată a unei clase, pointeri la membrii dată a unui obiect, membrii funcție, metode statice, metode inline, funcții membru cu this constant, constructori, constructori pentru clase cu obiecte incluse, constructori privați, constructori implicați, constructori cu argumente, constructori pentru copierea obiectelor, destructori, destructori privați, relația friend
7. Moștenirea, moștenirea simplă, polimorfism, închidere funcțională, moștenire multiplă, clase de bază virtuale, regula de dominanță, operatorul de scope resolution
8. Supraîncărcarea, conversii utilizator prin supraîncărcare, supraîncărcarea operatorilor, operatorii ca și apeluri de funcții, operatori supraîncărcați ca și funcții membru, operatori supraîncărcați ca și funcții friend, operatorul apel de funcție, operatorul de subscriere (indexare), operatorul de asignare, limitări ale supraîncărcării operatorilor, operatori unari prefix și postfix, accesul membrilor clasă folosind ->, supraîncărcarea lui new și delete
9. Polimorfism, early and late binding, funcții virtuale, funcții virtuale nule, clase abstracte, tipuri de funcții virtuale, vptr și vtab
10. Streams, fișiere, formatare, filtre
11. Biblioteca de șabloane standard (STL), containeri, iteratori, traversatori și predicate, algoritmi