



UNIVERSITÀ  
DEGLI STUDI DI MILANO-BICOCCA

## COURSE SYLLABUS

### C++ Programming

2425-3-E3101Q133

---

#### Obiettivi

Al termine del corso, lo studente sarà in grado di progettare e sviluppare programmi modulari e manutenibili. Sarà inoltre in grado di applicare le moderne tecniche di programmazione C++ per sviluppare applicazioni performanti e grafiche gestendo in modo corretto ed attento le risorse. Le competenze acquisite consentiranno allo studente di affrontare e comprendere le applicazioni C++ più complesse.

#### Contenuti sintetici

Il corso si prefigge di dare allo studente le basi necessarie per affrontare lo sviluppo di applicazioni C++ in modo corretto ed attento alle problematiche di gestione delle risorse. A tal fine verranno mostrate, attraverso l'uso intensivo di casi di studio, le criticità e difficoltà inerenti al linguaggio C++ e le tecniche più adatte ad affrontarle. Verrà inoltre presentato un framework cross-platform per lo sviluppo di applicazioni C++ grafiche.

#### Programma esteso

Introduzione al C++.

Concetti base di programmazione C++

- tipi di dati, puntatori, reference, scoping
- casting,

C++ come linguaggio ad oggetti

- classi, costruttori e distruttori, overloading, metodi friend
- inline, constness"

Concetti avanzati di programmazione C++

- overloading degli operatori
  - metodi virtual, abstract, polimorfismo
  - ereditarietà
- Programmazione generica
- template
  - iteratori

#### La libreria Standard (STL)

- Le classi container
- Gli algoritmi
- Funtori
- Multithread

#### Uso delle librerie esterne

- Librerie statiche
- Librerie dinamiche
- La libreria OpenMP

#### I nuovi standard C++11, C++14

#### Applicazioni GUI

- Ambiente di sviluppo QT Creator
- Sviluppo di interfacce grafiche
- Gestione degli eventi
- Le librerie Qt, QWidgets

## Prerequisiti

Conoscenze di base di programmazione.

## Modalità didattica

Insegnamento erogato in lingua italiana.

L'insegnamento è strutturato nel seguente modo:

48 ore di lezioni frontali in modalità erogativa ed interattiva in presenza

20 ore di esercitazioni in modalità erogativa ed interattiva in presenza

## Materiale didattico

Bjarne Stroustrup, The C++ Programming Language - Special Edition, Addison Wensley. La "Special Edition" è la versione riveduta e corretta della "Third Edition".

Bruce Eckel, Thinking in C++ vol. 1 e vol. 2, Prentice Hall (disponibile online)

Peter Van Weert, Marc Gregoire, C++ Standard Library Quick Reference, Apress

Lee Zhi Eng, Qt5 C++ GUI Programming Cookbook, Packt Publishing

Slides e dispense varie.

## **Periodo di erogazione dell'insegnamento**

III° anno, primo semestre

## **Modalità di verifica del profitto e valutazione**

La verifica dell'apprendimento comprende dei compitini, una prova progettuale ed una prova orale.

Sono previste due prove in itinere (copitini) con domande di teoria inerenti agli argomenti del corso affrontati fino a quel momento.

Il compitino contiene 15 domande a risposta libera lunga a scelta multipla.

Il compitino è superato se le risposte ritenute corrette sono 11 su 15.

Ciascuna prova in itinere, se superata, garantisce 2/30 punti nella valutazione finale e sostituisce le domande di teoria all'orale su quegli argomenti.

La prova progettuale prevede lo sviluppo di una soluzione ad un problema assegnato con le tecniche e gli strumenti visti nel corso. Il problema è definito in modo tale da verificare l'acquisizione delle competenze pratiche e di problem solving. Il testo del progetto è pubblicato circa tre settimane prima dell'appello d'esame e gli studenti hanno due settimane per la consegna. La valutazione massima del progetto è 26/30. Alcuni criteri sulla valutazione del progetto sono:

- Se in seguito a dei test effettuati dai docenti in fase di valutazione (es. chiamate a funzioni non testate da voi), il codice non compila, l'esame NON è superato.
- Implementazione di codice non richiesto non dà punti aggiuntivi ma se non corretto penalizza il voto finale.
- Gli errori riguardanti la gestione della memoria sono considerati GRAVI.
- La valutazione del progetto non dipende dalla quantità del codice scritto.

La prova orale consiste in una discussione della soluzione sviluppata, domande di teoria sulle nozioni presentate a lezione (in caso di non superamento dei compitini) e domande di ragionamento e deduzione. La valutazione della prova orale, consente di aumentare (o diminuire) la valutazione base ottenuta in precedenza nel progetto.

## **Orario di ricevimento**

Su appuntamento

## **Sustainable Development Goals**

