

Materie informatiche – Programmazione **Programmazione OO**

Obiettivi

Sapere

Concetti fondamentali della concezione OO, quali: classi, oggetti, ereditarietà, polimorfismo e "dynamic binding"

Gli elementi fondamentali dei linguaggi OO C++ e JAVA

Saper fare

Modellare il mondo reale tramite Classi e Oggetti

Definire e implementare una Classe in C++ e JAVA

Identificare diverse scomposizioni modulari per un problema dato, valutarle e scegliere la più adatta

Utilizzare un ambiente di programmazione per C++ e per JAVA (librerie di funzioni standard)

Saper essere

Analizzare un problema, scomporlo in unità logiche (moduli), e riconoscere gli oggetti

Codificare, nei linguaggi C++ e JAVA una o più classi nell'ambito dei lavori di gruppo

Dimostrare un atteggiamento positivo nel lavoro in seno a un gruppo di sviluppo

Programma

Dalla programmazione strutturata alla concezione a oggetto

Introduzione:

"Crisi del Software" e ingegnerizzazione

Fattori di qualità

Astrazione dei dati

La protezione dei dati:

Divisione di programmi

Modellazione attraverso i tipi di dati

Località e "scope"

La modularità:

Generalità

Criteri e principi di modularità

La programmazione a oggetti:

Generalità

Il concetto di classe e di oggetto

Il concetto di ereditarietà

Linguaggi dinamicamente tipati verso linguaggi staticamente tipati

I concetti di polimorfismo e "dynamic binding"

Un'introduzione al linguaggio C++

Introduzione

Le classi in C++

Funzioni membro e funzioni amiche

La sovraccarica dei nomi e degli operatori

I costruttori

Classi derivate Funzioni virtuali

Il polimorfismo e il "dynamic binding" in C++

Un'introduzione al linguaggio JAVA Introduzione

Lessico e sintassi del linguaggio

Confronto con C++

il JDK (Java Developing Kit)