

Introduzione alla programmazione in C++ / uso della memoria dinamica

Introduzione al C++ e chiarimenti : differenze fra C e C++ nell'I/O standard, riferimenti, operatore ::, allocazione/deallocazione dinamica con new/delete in alternativa a malloc/free/sizeof. Allocazione e deallocazione dinamica delle variabili semplici attraverso pointer (tipi base e tipi strutturati). Argomenti di default, sovraccarico (overload) e generalizzazione di funzioni.

Programmazione ad oggetti in C++

Scopo della programmazione orientata agli oggetti, definizione di classe, metodo, attributo, incapsulamento dei dati e visibilità. Vantaggi nell'approccio ad oggetti, sia nella semplicità di codifica, ma soprattutto nella fase di manutenzione e revisione del software.

Le classi in C++: attributi e metodi privati e pubblici, costruzione di semplici classi. Metodi per l'input e per l'output, costruttore e distruttore, sovraccarico (overload) dei metodi.

Overload del costruttore e richiamo implicito nella definizione dei dati.

Utilizzo del costruttore/distruttore con vettori. Vantaggi del costruttore/distruttore nell'allocazione dinamica. Il puntatore interno "this". Overload degli operatori in C++ : overload operatori matematici (+, -, *, /), overload operatori di I/O (>>, <<), overload operatori relazionali (>, <, >=, <=, ==), overload operatori incremento/decremento (++ , --), overload operatore di indicizzazione ([]); utilizzo dell'overload degli operatori per la semplificazione del codice ad alto livello e per la gestione di controlli di integrità dei dati.

Funzioni friend: definizione, utilità, problematiche, utilizzo. Overload di alcuni operatori come funzione friend. Ereditarietà (singola): ereditare da una classe, ereditare costruttori e metodi, costruire nuovi metodi attraverso l'uso di metodi ereditati, utilizzare metodi ereditati. Ereditarietà multipla. Introduzione al Polimorfismo: funzioni virtuali.

Programmazione ad Oggetti in Java e cenni di GUI

Introduzione al linguaggio:

Caratteristiche del linguaggio, differenze e similitudini rispetto al C++, Jre e Jdk e componenti, concetto di JVM, JDK, utilizzo e configurazione di notepad++, Bluej e VsCode per compilare, indentare in Java, tipi base del linguaggio, input da tastiera/file/numeri pseudocasuali, input ed output verso file di testo. Vettori ed algoritmi notevoli, funzioni statiche ed utilizzo, dinamicità dei vettori: dimensionamento e attributo length, arrotondamenti e formattazione dell'output. Stringhe in Java e principali metodi (contains, equals, split, replace, ...).

Programmazione ad oggetti in Java:

Oggetti ed esempio di classi: costruttori, getter e setter, attributi, oggetto interno (this).

Vettori di oggetti ed arrayList in Java; metodi principali del container arrayList (size, get, set, add, remove, ...), classi primitive e Wrapper corrispondenti (Integer, Double, ...), ereditarietà in Java.

Programmazione GUI e intercettazione di Eventi:

Introduzione alle AWT e realizzazione di semplici applicativi (counter) con Button, Label, TextField, ... pannelli e semplici layout. Uso del metodo split per sequenze di bottoni (Calcolatrice). Gestione di semplici eventi (click su bottone, click del Mouse).

Laboratorio

Utilizzo di ambiente IDE C/C++: CodeBlocks/DevCpp & Mingw , Notepad++, BlueJ, VsCode & Java, Utilizzo di Compilatori on line, appunti e software libero da sito: **fabriziosacco.it**

Esercizi di ripasso sui vettori e gli algoritmi notevoli

Temi individuali e a gruppo con problematiche inerenti l'utilizzo di vettori e matrici.

Temi individuali e a gruppo con problematiche inerenti l'utilizzo di file di testo, csv, html.

Esercizi sulla programmazione ad oggetti: costruzione completa di classi (temi di laboratorio individuali) anche con l'utilizzo di overload di operatori (C++).

Creazione di programmi in Java.

Gli insegnanti

Gli alunni
