

PROGRAMMA SVOLTO DI INFORMATICA

Docenti: Ciaschetti Gianfranco, Lorenzo Podda

MODULO 1: RIPASSO ARGOMENTI ANNO PRECEDENTE

Algoritmi e Programmazione in C++; Costrutti della programmazione strutturata; Funzioni; Array

MODULO 2: RECORD E TABELLE IN C++

Definizione di struttura e record; array di record (tabelle); simulazione delle operazioni per la gestione di un archivio con tabelle: inserimento, visualizzazione, ricerca, modifica, cancellazione.

MODULO 3: ARCHIVI E FILE IN C++

Definizioni di archivio e di file; file di testo e file binari in C++; accesso sequenziale e accesso casuale; operazioni fondamentali sui file in C++: apertura e chiusura di un file; lettura e scrittura su file di testo e file binari; accesso diretto e spostamento della testina di lettura/scrittura; fine del file; ricerca e modifica di un record in archivio; cancellazione logica e cancellazione fisica; realizzazione di un sistema di gestione di un archivio in C++ (inserimento, ricerche, accesso diretto, visualizzazione, modifica, cancellazione logica e cancellazione fisica).

LABORATORIO: esercitazioni sui file in C++; realizzazione di un archivio in C++.

MODULO 4: ALLOCAZIONE DINAMICA DELLA MEMORIA

Gli indirizzi della RAM; aritmetica dei puntatori; le istruzioni per l’allocazione dinamica in C++: *new* e *delete*; array dinamici e differenze con gli array statici; record dinamici.

LABORATORIO: esercitazioni sull’allocazione dinamica della memoria.

MODULO 5: TIPI DI DATO ASTRATTO (ADT)

ADT pila: logica LIFO, operazioni *push* e *pop* con implementazione statica e implementazione dinamica; ADT coda: logica FIFO, operazioni *push* e *pop* con implementazione statica e implementazione dinamica; ADT lista lineare o collegata: inserimento, ricerca, modifica, cenni sulla cancellazione e sull’inserimento ordinato (strategia *look ahead*); varianti della lista collegata: lista circolare, lista *double-linked*; visita degli elementi di una lista con logica iterativa e logica ricorsiva; ADT albero binario: visita in pre-ordine, in-ordine, post-ordine, ampiezza, inserimento, ricerca; alberi binari di ricerca (ABR): inserimento e ricerca; il concetto di bilanciamento di un albero: bilanciamento perfetto e alberi bilanciati in altezza (AVL).

LABORATORIO: esercitazioni sugli ADT pila, coda, lista lineare, lista doppiamente collegata, lista circolare, albero binario, albero binario di ricerca

MODULO 6: GRAFI E ALBERI

Definizione di grafo e albero; definizioni e terminologia: adiacenza, incidenza, cammini, percorsi, cicli, padri, figli, antenati, discendenti, profondità, livelli; memorizzazione di grafi: matrici di incidenza, matrici di adiacenza, cenni sulle liste di adiacenza; memorizzazione di alberi: puntatori ai figli, vettore heap, cenni sugli altri metodi di memorizzazione di un albero (puntatori a figli e fratello prossimo, liste di figli, puntatori a figli e padre).

LABORATORIO: esercitazioni sulla rappresentazione di grafi e alberi e sugli algoritmi di visita di un albero.

MODULO 7: PROBLEMI E ALGORITMI DI OTTIMIZZAZIONE SU GRAFO

Problema del Cammino minimo: definizione, applicazioni, algoritmo di Dijkstra;

Problema del Minimo albero ricoprente: definizione, applicazioni, algoritmo di Kruskal;

Problema della Colorazione dei vertici: definizione, applicazioni; casi particolari del problema della colorazione: cicli pari, cicli dispari, sottografi, grafi completi, grafi planari, grafi bipartiti

Problema dell'Abbinamento ottimo: definizioni di matching massimo e massimale, matching di peso massimo, matching completo; applicazioni e casi particolari: grafo bipartito, problema dell'assegnamento; algoritmo ungherese per il problema dell'assegnamento.

Problema dello zaino (cenni): formulazione, complessità, casi particolari, algoritmi risolutivi

Problema del commesso viaggiatore (cenni): formulazione, complessità, casi particolari.

Cenni sulla complessità computazionale di problemi e algoritmi di ottimizzazione.

LABORATORIO: progettazione e sviluppo di semplici algoritmi di ottimizzazione su grafi

MODULO 8: PROGRAMMAZIONE ORIENTATA AGLI OGGETTI

Modellazione a oggetti: dominio dell'applicazione, *object oriented analysis*, modello E/R e diagramma delle classi, UML: Unified Modeling Language; entità: attributi e metodi; relazioni ISA, HAS, relazioni di associazione generica; molteplicità delle relazioni; dal diagramma E/R al programma in C++; creazione e distruzione di oggetti; accesso ad attributi e metodi di oggetti statici e dinamici; metodi costruttori e distruttori; caratteristiche della programmazione a oggetti: ereditarietà (superclassi e classi derivate), incapsulamento (regole di visibilità, metodi *get* e *set*, separazione tra interfaccia e implementazione, funzioni friend, la clausola *this*), polimorfismo (*overloading* e *overriding* di metodi, namespace, *late binding* e metodi virtuali, *overloading* di operatori).

LABORATORIO: analisi, progettazione e modellazione delle classi con LucidChart; sviluppo di applicazioni OOP, anche in collaborazione con altre discipline (TPSIT), con DevC++ o Visual Studio Code; Sviluppo di semplici applicazioni OOP in Java in modalità “console” e con NetBeans.

MODULO 9: INTRODUZIONE ALLE INTERFACCE UTENTE GRAFICHE

Esempi di semplici applicazioni con interfacce grafiche in Java (NetBeans) e in C# (Windows Forms).

LABORATORIO: realizzazione di semplici applicazioni con interfacce grafiche in Java con NetBeans e in C# con Windows Forms e Microsoft Visual Studio Community.

Carbonia, 6 Giugno 2025

I Docenti:

Gianfranco Ciaschetti

Lorenzo Podda