

1^A QUADRIMESTRE:

Introduzione all'informatica ed all'utilizzo di un sistema di sviluppo del software

Concetti di Hardware/Software. Concetto di periferiche e principali periferiche : classificazione in base all'input/output. Utilizzo di un sistema di sviluppo con ambiente integrato: CodeBlocks e DevCpp/Mingw. Compilazione del codice sorgente: identificazione e gestione degli errori, introduzione al debugging e all'utilizzo del compilatore. Configurazione dell'IDE (DevCpp) per la memorizzazione di Codice/completamento automatico.

Fondamenti di programmazione

Schematizzazione di un algoritmo utilizzando la tecnica del diagramma di flusso; simboli usati nei diagrammi (rettangolo per le istruzioni, parallelogramma per l'I/O, rombo per le condizioni logiche booleane, ellissoide per inizio/fine) sequenzialità, alternativa e iterazione. Introduzione alla programmazione tramite diagramma di flusso (uso di flowgorithm).

Concetti di variabile, Input/Output e sequenzialità:

Variabili e costanti con preprocessore, variabili base in C e range di valori possibili.

Variabili con segno e senza segno in aritmetica intera (char, short, int, long) con e senza segno (unsigned). Variabili in virgola mobile (float, double). Input/Output standard da tastiera (printf, scanf, getchar, putchar), da file (fprintf/fscanf/getc/putc)

Linguaggio di programmazione e codifica

Concetto di esecutore, compilatore, programmatore, utilizzatore. Codifica passo a passo di semplici programmi contenenti istruzioni in sequenza, condizioni logiche e scelte alternative, cicli. Realizzazione di programmi funzionanti dopo aver progettato l'algoritmo risolutivo e disegnato il diagramma di flusso.

Istruzioni decisionali

If else, if annidati e composti: esempi ed applicazioni, risultato di espressioni booleane e loro utilizzo. Logica booleana del C ed applicazioni anche con varie condizioni. Esempi ed applicazioni.

Istruzioni iterative:

Istruzione WHILE (controllo in testa), istruzione FOR e sue varianti, Istruzione DO - WHILE (controllo in CODA). Cicli annidati e controllo degli indici. Esempi ed Applicazioni.

2^A QUADRIMESTRE:

Vettori e Matrici e algoritmi notevoli:

Array e loro utilizzo: inizializzazione, I/O standard, algoritmi notevoli: minimo, massimo, media, ricerca lineare, ordinamenti per scambio e per selezione e per tabella. Ordinamenti avanzati: algoritmo di quicksort. Utilizzo di vettori per rappresentazione di semplici modelli della realtà.

Array bidimensionali (matrici): costruzione, rappresentazione ed utilizzo. Gestione per linea e per colonna ed applicazione alle matrici di alcuni dei principali algoritmi visti sui vettori.

Stringhe: rappresentazione interna in C. Semplici funzioni di base per l'I/O standard e per la gestione delle stringhe; funzioni di libreria standard strcpy(), strlen(), strcmp(), strcat(), etc. e loro ricodifica sia con vettori che con puntatori.

Modularità del software ed uso di funzioni:

Introduzione graduale alla modularità ed alla documentazione del software: concetti di sottoprogramma, passaggio dei parametri per valore e per indirizzo: particolarità in C. Simulazione del passaggio per indirizzo mediante pointer in C standard, valori di ritorno e prototipo delle funzioni. Funzioni annidate e loro utilizzo e gestione. Riutilizzabilità del codice con particolare riferimento alla costruzione e documentazione del software.

File (sequenziali, modalità testo):

Concetto di file, apertura, lettura/scrittura e chiusura file. Funzioni standard per la lettura/scrittura. Elaborazione di file di testo con output su video e su file.

Laboratorio e sviluppo software:

Sviluppo di software e documentazione del codice con stesura di programmi e funzioni di difficoltà via via crescente

Quattro operazioni con due numeri introdotti da tastiera, tabelline.

Gestione di dati in vettori e semplici algoritmi: calcolo minimo, massimo, media, ordinamento ed altre elaborazioni su vettori, utilizzando funzioni.

Costruzione di funzioni per la gestione di stringhe e ricodifica di alcune funzioni di libreria standard.

Produzione di output su schermo, su file di testo e su file HTML.

Utilizzo della rete per il download di esercizi, documentazione e l’upload di compiti e lavori di laboratorio.

Materiale didattico: appunti, documenti, lezioni, IDE e software freeware, preparati ad hoc dagli insegnanti e scaricabili dal sito: fabriziosacco.it.

Gli insegnanti

Gli alunni
