

Programma del corso di Linguaggi di Programmazione, corso di laurea in Informatica e Comunicazione Digitale- sede di Taranto. A.A. 2004-2005 (secondo semestre)

1. Introduzione al Corso

Richiami su problemi, algoritmi e programmi. Progettazione di un algoritmo. Linguaggi lineari. Il linguaggio Pascal. Il processo di traduzione.

2. Sintassi dei Linguaggi di Programmazione

Sintassi e Semantica. Grammatiche. Classi di grammatiche di Chomsky. Backus-Naur Form (BNF). Alberi Sintattici. Analizzatore sintattico. Analizzatore Lessicale (scanner). Analizzatore Grammaticale (parser). Linguaggi e Automi.

3. Semantica dei Linguaggi di Programmazione

Introduzione. Semantica Operazionale. Semantica Denotazionale. Semantica Denotazionale Funzionale. Semantica Denotazionale Assiomatica.

4. Correttezza dei programmi

Le asserzioni. Regole dell'assegnamento, della sequenzialità, della composizione, del condizionale e del while. Correttezza dei programmi con procedure. Correttezza dei programmi ricorsivi. Terminazione dei programmi.

5. I paradigmi di programmazione

Definizione. Approccio Operazionale e Definizionale. Rapporto Linguaggi/Paradigmi.

6. Il paradigma Imperativo

Architettura di von Newmann. Concetti Base. Il paradigma imperativo. I linguaggi imperativi. I dati: nome, rappresentazione e tipo. Dichiarazione di Dati. Tipi primitivi, composti e ricorsivi. Le espressioni. I comandi. La gestione del flusso di controllo. La composizione (blocchi e sottoprogrammi). Legame statico e dinamico. Procedure e Funzioni. Passaggio di parametri.

Linguaggi di programmazione di riferimento.

Linguaggio C, Linguaggio Ada

7. Astrazione

L'astrazione nella programmazione e il principio di Astrazione. Astrazione di funzione, di procedura, di controllo e di selettore. Astrazione di tipo e tipi astratti di dato. Specifiche algebriche e assiomatiche per i tipi astratti di dato. I moduli per l'incapsulamento dell'informazione e l'information hiding. Oggetti e classi di oggetti. Astrazione di dati: Tipo astratto di dato vs. classe di oggetti. Astrazione generica.

Linguaggi di programmazione di riferimento.

Linguaggio C, Linguaggio Ada, I moduli in Modula-2, Turbo Pascal.

8. Il paradigma Orientato agli Oggetti

Introduzione. Le classi e le istanze di classe. Variabili d'istanza e variabili di classe. Metaclassi. Ereditarietà singola ed ereditarietà multipla. Gerarchia di classi e grafo di ereditarietà. Classi Astratte. Composizione di oggetti. Confronto tra ereditarietà e composizione. Tipizzazione statica e dinamica. Legame statico e dinamico. Polimorfismo del software e classificazione di polimorfismo. Coercizione, overloading, polimorfismo parametrico e polimorfismo per inclusione.

Linguaggi di programmazione di riferimento.

Linguaggio C++.

Esercitazione

1. Il Linguaggio C

Genealogia del Linguaggio C. Relazioni con altri linguaggi. Vantaggi e Svantaggi del C. Oggetti, variabili e tipi. Gli identificatori. Tipi di dati standard. Enumerazioni. Modificatori di accesso. Conversioni di tipo. Dichiarazione di variabili. Classi di memorizzazione. Dichiarazione di funzioni. Gli Operatori. Operazioni di I/O. Controllo condizionale e iterativo. Le funzioni: chiamata per valore e per indirizzo. La ricorsione. Gli argomenti delle funzioni. Il tipo delle funzioni.

2. I Dati Strutturati

Gli array. Array multidimensionali. I puntatori. Puntatori ad array. Vettori di puntatori. Puntatori a puntatori. Puntatori a funzioni. Allocazione dinamica della memoria. Le strutture (*struct*). Passaggio di strutture. Unioni. Macro.

3. Implementazione degli algoritmi fondamentali in C

Partizionamento di un array. Select sort. Bubble sort. Insertion sort. Shell sort. Diminuzione di incremento. Quick sort. Fusione (merge). Merge sort. Ricerca binaria.

4. Programmare in O-O: C++

Miglioramenti rispetto al C. Dalle strutture alle classi. Classi e oggetti. Puntatori e riferimenti. Ereditarietà. Funzioni virtuali. Ereditarietà multipla. Membri pubblici e privati. Template. Espansione in linea. Overloading degli operatori. Funzioni operatore friend. Librerie standard del C++.

5. Astrazione Dati in C/C++

Tipo astratto *Lista* (realizzazione con cursori, puntatori). Tipo astratto *Pila* (realizzazione con vettore, puntatori). Tipo astratto *Coda* (realizzazione con vettore circolare e con puntatori)

Materiale di Riferimento:

Dispense del corso.

M. Maiocchi. (parte 1., 2.)

Teoria e applicazioni delle macchine calcolatrici. cap. 1,2,3.

Casa editrice Ambrosiana. (1990)

C. Batini, L. Carlucci Aiello, M. Lenzerini, A. Marchetti Spaccamela e A. Miola (parte 2., 3. e 4.)

Fondamenti di programmazione dei calcolatori elettronici, cap. 1 e 5

ed. Franco Angeli, (2000).

Wirth (parte 1. e 4.-terminazione)

Principi di programmazione strutturata cap. 5., 6., 7

ISED I 1999

A. Andronico (parte 4.)
Fondamenti dell'Informatica cap. 3
Zanichelli

A.L. Ambler, M.H. Burnett, & B.A. Zimmerman (parte 5.)
Operational Versus Definitional: A Perspective on Programming Paradigms
IEEE Computer, 25(9): 28-43, September 1992.

H. E. Bal, D. Grune (parte 6.)
Programming Languages Essentials. cap. 1-2
Addison-Wesley, 1994

E. Lodi, & G. Pacini (parte 7.)
Introduzione alle Strutture di Dati (cap. 3-4)
Bollati Boringhieri, 1990.

F. Crivellari (parte 7. e 8.)
Programmazione ad Oggetti e tipi di dati astratti.
(1997)

R.G. Dromey (parte Esercitazione_3.)
Algoritmi fondamentali
Jackson.

A. Bertossi. (parte Esercitazione_5.)
Algoritmi e strutture dati,
UTET, 2000.