FONDAMENTI DI INFORMATICA

Domenico Talia
talia@deis.unical.it

A.A. 2003-2004
Facoltà di Ingegneria
UNICAL

Fondamenti di Informatica -Obiettivi

- L'obiettivo del corso e' lo studio delle metodologie di base della programmazione dei calcolatori e della loro applicazione nello sviluppo di moduli software in **Java** che utilizzino tipi di dati semplici ed array.
- Il corso introduce alle tematiche relative alla programmazione ad oggetti.
- Crediti : 4 CFU.

Fondamenti di Informatica - Orario

- Periodo: 12 Gennaio – 13 Marzo.
- Ogni settimana 3 ore di lezione e 2 di esercitazione.
- In totale 25 ore di lezione e 12 ore di esercitazioni.
- Ore aggiuntive di tutoraggio (opzionali)
- Ricevimento :
  Martedi 17:30-19:30
  DEIS, cubo 41c, 3° piano.

Fondamenti di Informatica - Programma

Introduzione alla programmazione e all'organizzazione dei calcolatori

- Risoluzione algoritmica dei problemi. Correttezza ed altre proprietà degli algoritmi.
- Algoritmi e programmi. Livelli di astrazione e linguaggi.
- Codifica binaria.
- La rappresentazione dell'informazione all'interno dei calcolatori: caratteri, numeri naturali, interi, reali.
- Algebra di Boole.

Programmazione su tipi semplici

- Introduzione alla programmazione orientata agli oggetti.
- Codifica di algoritmi in programmi Java.
- Struttura di un programma: costanti, variabili, tipi, classi, oggetti, costruttori.
- Istruzioni semplici e tipi predefiniti. Compatibilità di tipo nella assegnazione.
- Operazioni di ingresso/uscita.
- Istruzioni per il controllo del flusso di elaborazione.
- Sviluppo incrementale di programmi.

Metodi e programmazione con array

- Concetto di funzione e procedura.
- Metodi in Java.
- Esecuzione di metodi e passaggio dei parametri.
- Il costruttore di tipo array.
- Tipi array monodimensionali,
- Tipi array multidimensionali, manipolazione di array.
- Gestione di vettori e matrici.

Tecniche di programmazione

- Proprietà delle classi e degli oggetti.
- Semplici algoritmi di ricerca.
- Tecniche di ordinamento di vettori.
- CENNI di
  - Gerarchia di classi.
  - Classi per la gestione di file.
  - Classi per la gestione di vettori e stringhe.

- Ludici delle lezioni e programmi svolti nelle esercitazioni.
- Sito web :

  si.deis.unical.it/~talia/aa0304/fond.html

  con i lucidi in formato PDF.
- Materiale disponibile anche su iCampus (icampus.deis.unical.it) e ecampus.mat.unical.it
- CD-ROM

  L'ambiente JDK, esempi di programmazione, libreria di input e altro materiale.

Libri su Java

- M. Bertacca, A. Guidi, Introduzione a Java, McGraw-Hill.
- L. Cabibbo, Fondamenti di Informatica – Oggetti e Java, McGraw-Hill, 2004.
- K. Arnold, J. Gosling, Java – Didattica e Programmazione, Addison-Wesley.
- J. Hubbard, Programmare in Java, McGraw-Hill Libri Italia.

- L. Lemay, R. Cadanhead, Java 2 – Guida Completa, Apogeo.
- C.S. Horstmann, Concetti di Informatica e Fondamenti di Java 2, Apogeo.
- C.T. Wu, Introduzione alla Programmazione a Oggetti in JAVA, McGraw-Hill, 2000.
- Deitel & Deitel, Java Fondamenti di Programmazione, Apogeo, 2000.

Consultazione e approfondimenti

- D. Sciuto, ed altri, Introduzione ai sistemi informatici, McGraw-Hill.
- S. Ceri, D. Mandrioli, L. Sbattella: Istituzioni di informatica, McGraw-Hill.
- S. Ceri, D. Mandrioli, L. Sbattella: Informatica: arte e mestiere, McGraw-Hill.

Fondamenti di Informatica – Modalità di esame

- Prerequisiti : l'esame può essere sostenuto da chi ha superato l'esame di Introduzione all'Informatica.
- L'esame prevede lo svolgimento di
  - Una prova scritta di programmazione Java in aula con possibilità del voto massimo e
  - Una prova orale facoltativa sugli argomenti del programma con la possibilità di miglioramento del voto dello scritto.
  - La prova orale è obbligatoria per chi riporta un voto allo scritto compreso nell'intervallo chiuso (15,17).

Richiami di concetti di base: Calcolatore

- **Calcolatore Elettronico - Computer**:
  - Strumento per la rappresentazione e l'elaborazione dell'informazione *oppure*
  - Esecutore di algoritmi.
- **Il Calcolatore**
  - è uno strumento in grado di eseguire insiemi di azioni ("*mosse*") elementari
  - le azioni vengono eseguite su oggetti (**dati**) per produrre altri oggetti (**risultati**)
  - l'esecuzione di azioni viene richiesta all'elaboratore attraverso frasi scritte in un qualche linguaggio (**istruzioni**).

- **Concetto di Algoritmo**

  Sequenza finita di passi che portano alla realizzazione di un compito.

- **Proprietà fondamentali**
  - **Eseguibilità**: ogni azione deve essere eseguibile da parte dell'esecutore dell'algoritmo in un tempo finito;
  - **Non-ambiguità**: ogni azione deve essere univocamente interpretabile dall'esecutore;
  - **Finitezza**: il numero totale di azioni da eseguire, per ogni insieme di dati di ingresso, deve essere finito.

Richiami di concetti di base: Algoritmo

Quindi un algoritmo deve:

- Essere applicabile a qualsiasi insieme di dati di ingresso appartenenti al dominio di definizione dell'algoritmo;
- Essere costituito da operazioni appartenenti ad un determinato insieme di operazioni fondamentali;
- Essere costituito da regole non ambigue, cioè interpretabili in modo univoco qualunque sia l'esecutore (persona o "macchina") che le legge.

Altre proprietà desiderabili

- generalità, determinismo, efficienza.

Programmi e Programmazione

- **Programmazione**

  È l'attività con cui definiscono le operazioni che servono a predisporre l'elaboratore ad eseguire un particolare insieme di azioni su particolari dati, allo scopo di risolvere un problema.

- **Programma**

  Sequenza di istruzioni di un linguaggio di programmazione comprensibile al calcolatore che realizzano un compito o risolvono un problema.

Cosa è la Programmazione

- La programmazione è
  l'attività di progettare e realizzare una programma, cioè
  definire le istruzioni che indicano ad un calcolatore i passi da eseguire per risolvere un problema.

- Usare un computer non necessariamente richiede una attività di programmazione.

- Tuttavia imparare a programmare un computer è una delle attività principali in informatica ed è utile a progettare e realizzare soluzioni a problemi in numerosi settori.

- **PROGRAMMA** : È la descrizione di un algoritmo in un particolare linguaggio di programmazione.
  - Quali "parole chiave" ?
  - Quali dati ?
  - Quali operazioni elementari ?
  - Quali meccanismi di combinazione ?
- **Un linguaggio di programmazione** è una notazione formale per descrivere algoritmi che è comprensibile ad un calcolatore.

SINTASSI e **SEMANTICA**

- Ogni linguaggio è caratterizzato da:
  - **sintassi**: l'insieme di regole formali per la scrittura di programmi in quel linguaggio, che dettano le modalità per costruire frasi corrette nel linguaggio stesso.
  - **semantica**: l'insieme dei significati da attribuire alle frasi (sintatticamente corrette) costruite nel linguaggio.

- Una frase può essere sintatticamente corretta e tuttavia non aver alcun significato!
- Lo stesso può accadere per una istruzione.

- **Tipi di Linguaggi di Programmazione**
  - Linguaggi macchina e linguaggi assembler

    ogni azione è indicata in codice binario o con operazioni molto semplici e "rudimentali" : `ADD X, Y` oppure `STORE A`
  - Linguaggi imperativi (PASCAL, FORTRAN, C, BASIC, …)

    le azioni da compiere sono indicate in una sequenza che partendo dai dati si completa calcolando i risultati :

```
if a > 0 print ("valore positivo")
 else print ("valore negativo");
```

  - Linguaggi dichiarativi (logici - PROLOG, funzionali - LISP)

    un programma è la definizione di una funzione o l'elenco delle regole logiche che portano a verificare una condizione.

Programmi e Programmazione

- Linguaggi orientati agli oggetti (C++, Java, Smalltalk, ….)
  - Sono basati sul concetto di **oggetto** software che rappresenta un oggetto del mondo reale (un numero, un archivio, un testo, una matrice).
  - I dati sono rappresentati come oggetti e le azioni da compiere come operazioni da effettuare sugli oggetti.
  - Di solito sono realizzati come estensione dei linguaggi imperativi.
  - Un programma modella un problema reale come una collezione di oggetti software che interagiscono.

Programmi e Programmazione

- Per far eseguire un programma ad un calcolatore occorre tradurlo dal linguaggio usato nel linguaggio macchina.
- La traduzione avviene secondo due modalità principali:
- **Compilazione**
  - Il compilatore controlla che tutte le istruzioni del programma siano corrette e alla fine di questo controllo **se non ci sono errori** genera il programma eseguibile che verrà eseguito dall'**esecutore**.
- **Interpretazione**
  - L'interprete controlla una per volta ogni singola istruzione del programma e se questa è corretta **la traduce e la esegue**. **Al primo errore termina l'esecuzione del programma.**