

INSEGNAMENTO DI: FONDAMENTI DI INFORMATICA C

Docente:

Prof. Giacomo Cabri

Esercitatore:

Ing. Francesco De Mola

Come Contattarmi:**E-mail (consigliato)**

giacomo.cabri@unimore.it

Telefono

059-2056190

Ricevimento

Lunedì pomeriggio dalle 15 alle 17

Dipartimento di Ingegneria dell'Informazione

Via Vignolese 905, primo piano

Materiale sul Corso:

<http://www.agentgroup.unimo.it/fiC>

Libro del corso:

G. Cabri, F. Zambonelli, "Programmazione a oggetti in Java: dai fondamenti a Internet", Pitagora editrice, 2003

PROGRAMMA DEL CORSO

Programmazione ad Oggetti

Verranno illustrati i concetti generali, quali incapsulamento, ereditarietà e polimorfismo; verranno accennati i concetti di riusabilità e di composizione dei componenti software, illustrando il passaggio dalla programmazione modulare alla programmazione ad oggetti.

Il linguaggio Java

Verrà utilizzato come esempio di linguaggio ad oggetti, spiegando come i concetti generali sono implementati in Java; verrà inoltre spiegato come Java può essere sfruttato per lo sviluppo di interfacce grafiche.

Dati strutturati in Java

Verrà illustrata l'implementazione di strutture dati complesse come liste, pile, code, alberi e grafi in Java; su tali strutture verrà introdotto il problema della complessità e il relativo studio.

REGOLE ESAME

Discussione di un progetto realizzato in Java che utilizza:

- **Interfaccia grafica**
- **Caratteristiche della programmazione ad oggetti**
 - **Incapsulamento**
 - **Ereditarietà (progettata per il programma)**
 - **Polimorfismo**
- **Una struttura dati tra quelle viste a lezione**
- **I/O Java**

La discussione verterà sullo sfruttamento delle caratteristiche della programmazione ad oggetti nel programma presentato

Il programma deve essere corredato da documentazione HTML (ad es. generata con Javadoc)

Il programma deve essere eseguibile da riga di comando (NO ambienti di sviluppo IDE)

Problemi DA EVITARE in sede di esame:

- **Il programma non funziona**
- **Manca qualcosa (di solito, la documentazione o le strutture dati)**
- **Il progetto è diviso su più programmi (eventualmente da concordare con il docente)**
- **L'ereditarietà è usata male**
- **“Non mi ricordo perché ho fatto il programma tempo fa”**

Consiglio: studiare POCO su libri e appunti e fare invece MOLTA pratica al calcolatore!!!!

PERCHÉ QUESTO CORSO?

Fondamenti A e B:

- acquisizione nozioni base di informatica
- strumenti per la definizione di semplici algoritmi per la elaborazione di semplici strutture dati

MA:

- simpaticamente, il mondo reale presenta strutture dati complesse e algoritmi complessi, in un ambiente in continua evoluzione.
- Il calcolatore, deve supportare l'uomo nella gestione delle problematiche del mondo reale!
 - Strutture dati complesse e dinamiche
 - Algoritmi complessi!

Dovremo quindi:

- acquisire nozioni e strumenti più avanzati di programmazione
- esercitarsi nella gestione di strutture dati complesse
- riuscire a valutare la complessità computazionale dei programmi
- imparare metodologie per la riduzione della complessità progettuale dei programmi!!!!

ESEMPI

Descrizione Struttura Organizzativa Aziendale

Che struttura dati ci serve per rappresentarla?

- Non certo vettori o stringhe... è una struttura dati complessa, un grafo di relazioni e responsabilità....
- Inoltre, è dinamica, perché i membri della organizzazione vanno vengono e cambiano ruolo, e cambiano in numero....

Che algoritmi bisogna definire per gestire tali strutture e i loro cambiamenti?

Sistema di gestione a “numeri delle code” in banca:

- Che struttura dati la può rappresentare?
- Come gestiamo l'arrivo di nuovi clienti e il fatto che un cliente venga servito, uscendo dalla fila?
- Come gestiamo il fatto che ci potrebbe essere una gestione dei clienti privilegiati nella lista?

Banche dati DNA:

- il DNA ha struttura elicoidale complessa
- come memorizziamo i dati del DNA?
- che algoritmi ci servono per elaborare i dati memorizzati ed estrarne informazioni?

ALLARGANDO LA VISIONE....

Informatica è la scienza della “elaborazione automatica dell’informazione”

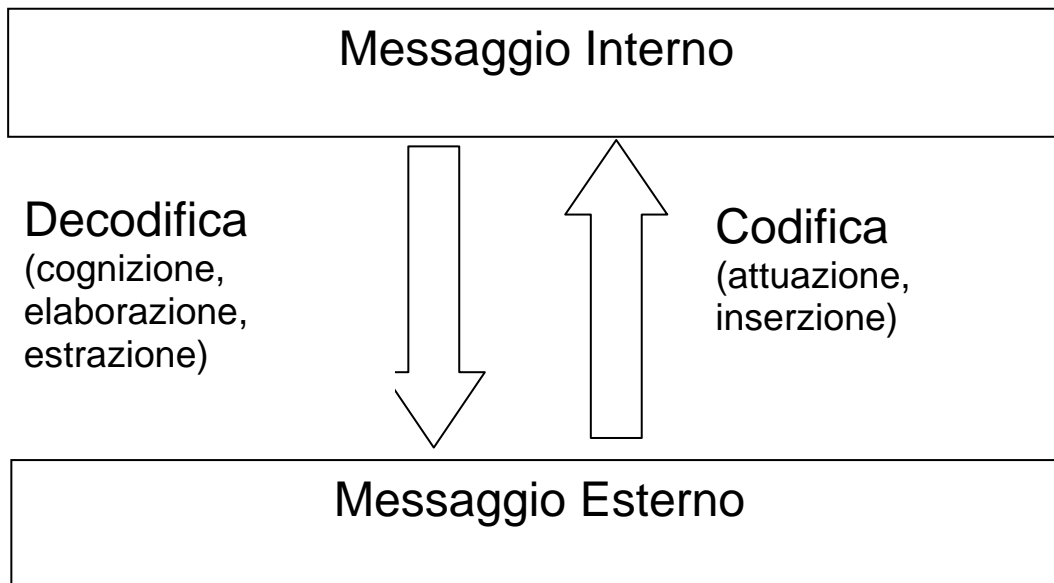
Il concetto di “elaborazione della informazione” è pervasivo nell’universo, e anche nelle nostre attività cognitive e attuative.

Il concetto stesso di informazione perde di senso senza gli algoritmi che la “elaborano”.

Esempi:

- **un pentagramma è informazione musicale? No, senza i processi cognitivi e attuativi che dalla partitura portano alla produzione di musica reale.**
- **il DNA è informazione su un essere vivente? NO, senza i processi bio-chimici che dal DNA formano un individuo.**
- **Un tuono è informazione su un temporale in arrivo? NO, senza i processi cognitivi che ci portano a dire “è solo il rumore di una scarica elettrica dovuta a una differenza di potenziale nell’atmosfera tipicamente associata a precipitazioni”**

IL CONCETTO DI INFORMAZIONE



L'informazione "utile" si produce realmente a livello di messaggio esterno.

Esempio:

DNA → messaggio interno

Processi chimico biologici → decodifica

Individuo → messaggio esterno

Riusciamo a capire come funziona? No, i messaggi interni e la decodifica sono troppo complicati!

Esempio:

$3x - 6 = 0$ → messaggio interno

regole algebriche → decodifica

$x = 2$ → messaggio esterno

IL MESSAGGIO INTERNO

L'informazione spesso è condivisibile, trasmissibile, o disponibile, solo a livello di messaggio interno.

Esempi:

- **Trasmissione caratteri ereditari → DNA**
- **Conoscenze matematiche → formule**
- **Memorizzazione stabile di musica nel 1800 → partitura**

Quindi:

Necessità di trasposizione di informazioni in una forma “forma interna” che ne faciliti la

- **trasmissione**
- **elaborazione**
- **ristrutturazione**

In ogni caso, motivi di economia intellettuale e fisica suggeriscono di:

- **minimizzare per quanto possibile la distanza tra la rappresentazione esterna e la rappresentazione interna e gli altri sforzi collegati!**

Esempio:

partitura musicale: astrazione delle corde del liuto

IL RUOLO DEI CALCOLATORI

Trasposizione della informazione relativa a oggetti/entità/strutture del mondo reale in una forma interna:

- **a basso livello: forma digitale**
- **ad alto livello: forma tipica del linguaggio di programmazione utilizzato**

Calcolatore come entità per:

- **elaborare dalla forma interna a quella esterna**
- **memorizzare**
- **trasmettere**
- **comprimere**
- **rendere disponibile in forma esterna**

Informazioni del “mondo reale”

O, più in generale:

per velocizzare ed estendere i processi cognitivi di elaborazione (codifica e decodifica) e memorizzazione dell'informazione che noi siamo comunque in grado di compiere!

IL RUOLO DEL PROGRAMMATORE

Nel suo lavoro di:

- **decidere come codificare informazione in forma interna**
- **definire “algoritmi” per la trasposizione dalla forma interna a quella esterna**

Deve avere come obiettivi:

- **minimizzare la distanza tra la forma interna e quella esterna (minimizzazione dello sforzo concettuale)**
- **minimizzare la complessità computazionale (sforzo algoritmico)**
- **minimizzare la complessità progettuale (sforzo di produzione del sistema software)**

Il buon programmatore lavora poco!

In verità, il lavoro è semplice, perché non rappresenta altro che una trasposizione dei nostri processi cognitivi e attuativi.....(per molti aspetti, noi siamo degli elaboratori!).

Scopo del corso, sotto questa luce:

- **acquisire le conoscenze “tecnologiche” necessarie per codificare ed elaborare le informazioni e le strutture del mondo, comunque complesse, in una forma interna elaborabile dal calcolatore**
- **studiare la codifica e la elaborazione di alcune strutture “pervasive”, quali le liste e le relazioni gerarchiche!**