

Fondamenti di Informatica

<http://www.gest.unipd.it/~info/info>

Alessandra Volpato

ale.volpato@ieee.org

INTRODUZIONE ALLA PROGRAMMAZIONE

Introduzione (1): cos'è un programma?

- ogni programma svolge una diversa funzione, anche complessa
 - ✓ impaginare testi, giocare a scacchi
- un computer è quindi una macchina che
 - ✓ Memorizza dati (numeri, parole, immagini, suoni,...)
 - ✓ Interagisce con dispositivi (schermo, tastiera, mouse,.....)
 - ✓ Esegue programmi
- i programmi sono sequenze di istruzioni che il computer esegue e di decisioni che il computer prende per svolgere una certa attività'

Introduzione (2): cos'è un programma?

- nonostante i programmi siano molto sofisticati e svolgano funzioni molto complesse, le istruzioni da cui sono composti sono molto elementari, ad esempio
 - ✓ estrarre un numero da una posizione della memoria
 - ✓ sommare due numeri
 - ✓ inviare la lettera A alla stampante
 - ✓ accendere un punto rosso in una data posizione dello schermo
 - ✓ se un dato è negativo, proseguire il programma da una certa istruzione anziché dalla successiva (decisione)

Introduzione (3): cos'è la programmazione?

- un programma descrive al computer, in estremo dettaglio, la sequenza di passi necessari per svolgere un particolare compito
- l'attività di progettare e realizzare un programma è detta *programmazione*
- usare un computer non richiede alcuna attività di programmazione (non occorre essere un meccanico per guidare l'automobile!!)

Introduzione (4): a cosa si applica la programmazione?

- quale tipo di problemi e' possibile risolvere con un computer?
 - ✓ Dato un insieme di fotografie di paesaggi, qual'e' il paesaggio piu' rilassante?
 - ✓ Avendo depositato 20000 euro in un conto bancario che produce il 5% di interessi all'anno, capitalizzati annualmente, quanti anni occorrono affinche' il saldo del conto arrivi al doppio della cifra iniziale?
- il primo problema non puo' essere risolto dal computer: perche'?

Introduzione (5): a cosa si applica la programmazione?

- il primo problema non puo' essere risolto dal computer perche' non esiste una *definizione* di **paesaggio rilassante** che possa essere usata per confrontare in modo *univoco* due paesaggi diversi.
- un computer puo' risolvere soltanto problemi che potrebbero essere risolti anche manualmente
- e' solo molto piu' veloce (e non si annoia...)

Introduzione (6): imparare a riconoscere i problemi

1. Dati due numeri, trovare il maggiore
2. Dato un elenco di nomi e relativi numeri di telefono (rubrica o elenco telefonico), trovare il numero di telefono di una persona
3. Data la struttura della rete stradale di una città e le informazioni sui flussi di veicoli in ciascuna strada ed in ogni momento della giornata, trovare il percorso a tempo minimo per andare dalla propria casa al lavoro
4. Decidere, per ogni x e per ogni funzione $f(x)$, se $f(x)$ è o meno una funzione costante.

I problemi trasformano un insieme di informazioni di ingresso, i dati, in nuove informazioni : i risultati

Introduzione (7): cos'hanno in comune i problemi

1. La descrizione non fornisce un metodo per calcolare il risultato
2. La descrizione del problema puo' essere ambigua o imprecisa
3. Per alcuni problemi non e' stato trovato un metodo risolutivo generale
4. Per alcuni problemi, il metodo risolutivo esiste in linea teorica, ma la sua applicazione puo' essere troppo lunga.

La prima cosa di cui preoccuparsi e' trovare almeno un procedimento risolutivo del problema, cioe' un insieme di regole che eseguite ordinatamente permettono di calcolare i risultati del problema a partire dai dati a disposizione 9

Introduzione (8): cos'hanno in comune i problemi

La prima cosa di cui preoccuparsi e' trovare almeno un procedimento risolutivo del problema, cioe' un insieme di regole che eseguite ordinatamente permettono di calcolare i risultati del problema a partire dai dati a disposizione

Questo procedimento e' detto **algoritmo**

Introduzione (8): proprietà di un algoritmo

1. Non ambiguità': le istruzioni devono essere univocamente interpretabili
2. Eseguibilità': le istruzioni devono essere eseguibili con le risorse a disposizione in un tempo finito
3. Finitezza: l'algoritmo deve terminare in un numero finito di passi per ogni valore dei dati di ingresso.

Un buon metodo di progettazione consiste nell'operare per livelli di astrazione (top-bottom), partendo da una versione iniziale molto generale per affinare successivamente i dettagli

Introduzione (9): scelta di un algoritmo

Numeri di telefono: come la rappresentazione delle informazioni puo' modificare l'algoritmo

- Un fattore determinante e' come sono rappresentabili o come scegliamo di rappresentare i dati in ingresso

Massimo comun divisore di due numeri n ed m

- Le potenzialita' espressive del linguaggio di programmazione svolgono un ruolo fondamentale

La somma dei primi n numeri interi

- L'efficienza di due algoritmi puo' essere molto diversa

Introduzione (10): risolvere un problema

- Individuare un algoritmo
 - Individuare una rappresentazione dell'algoritmo e delle informazioni
 - ✓ in ingresso
 - ✓ utilizzate dall'algoritmo nella sua esecuzione
- in un linguaggio di programmazione opportuno