

DEFINIZIONI DI BASE

Problemi computazionali

Esprime la relazione tra due insiemi:

- input (istanza): forma in cui si può rappresentare un problema (termini assoluti); contiene i dati.
- output (soluzione): è descritto in termini d'istanza; contiene i risultati.

Alcuni esempi sono l'ordinamento (input: sequenza disordinata, output: sequenza ordinata), la ricerca (input: parola chiave, output: risultato associato alla chiave), il prodotto di matrici (input: numero di matrici, output: il loro prodotto), l'involucro convesso (input: numero di punti, output: il più piccolo che li contiene).

Algoritmi

È una procedura che dall'input produce valori di output. È detto corretto quando esso termina e produce un output corretto per ogni istanza. Una procedura di calcolo ben definita ammette due tipi di risposte:

- una *risposta rigorosa*, cioè computabile con la RAM. Implementazione di dati con una RAM ha costi molto elevati e costituiscono linguaggi macchina elementari;
- una *risposta pragmatica*, con uno pseudocodice con certe regole che sostituisce la RAM.

Strutture di dati

sono strutture dove vengono salvati i dati. Hanno lo scopo di salvataggio temporaneo e "pre-processing" (prima dell'inizio della lavorazione) per rendere più facile la loro elaborazione. Queste strutture contengono dati per renderlo disponibile nella forma più utile e veloce.

Le **variabili** hanno associate un tipo ma non serve dichiararlo; sono denotate da un segno di eguaglianza e si possono fare assegnazioni multiple.

Gli **array** sono una sequenza di variabili di ogni tipologia.

Le istruzioni sono strutturate in **condizionali** (if), **booleane** (true & false), **ripetitive** (for, while, repeat until).

Gli **oggetti** sono dati strutturati in una serie di attributi definite da ("oggetto.attributo"); per porre a 0 un riferimento si usa la costante NIL.

Nelle **procedure** i parametri sono passati per valore, in cui la procedura chiamata riceve una copia dei parametri locali e se si modifica non ha effetto all'esterno. Se il parametro è un oggetto viene riprodotta solo una copia del riferimento all'oggetto. Le variabili sono locali alle procedure a meno che non sia indicato diversamente. Le procedure riportano un solo valore (per riportare più valori si fa uso degli oggetti).

Una funzione è una porzione di codice che produce un valore di ritorno.

Generalmente esse operano su variabili globali; i parametri formali sono inizializzati con i valori dei parametri attuali al momento della chiamata.

Passaggio per valore: le modifiche effettuate sui parametri attuali non sono visibili dal chiamante.

Passaggio per variabile: sono visibili dal chiamante (viene passato l'indirizzo di memoria della variabile e copiato nell'altra e quindi si modifica quello iniziale).

ESERCIZI

1) Scrivere una procedura in pseudocodice che calcoli il massimo tra due interi.

Massimo (x,y)

if (x>y) then

return x

//esci subito quando ci sta il return

return y

2) Scrivere una procedura in pseudocodice che prenda in input un array di interi e calcoli la somma dei valori contenuti.

Somma (a)

tot = a[0]

for i = 1 to a.length-1

tot = tot + a[i]

return tot

3) Scrivere una procedura che verifichi che un array di interi contenga solo valori maggiori di zero.

Verificapositivi (a)

positivi = true

//true perchè sto facendo ...

for i = 0 to a.length-1

// ... una verifica esistenziale

if a[i] < 0 then

positivi = false

return positivi

4) Scrivere una procedura in pseudocodice che prenda in input un array di interi e produca in output il valore massimo contenuto e la sua posizione nell'array.

Massimo (a)

x.massimo = a[0] //massimo e pos sono i campi di x

x.pos = 0

for i = 1 to a.length-1

if a[i] > x.massimo then

x.massimo = a[i]

x.pos = i

return x