

Algoritmi e Programmi

Dato un problema, per arrivare ad un programma che descriva la soluzione dobbiamo:

- Definire il risultato da ottenere: le uscite (output);
- Individuare gli elementi a disposizione: gli ingressi (*input*);
- Trovare un metodo di risoluzione che include:
 - un algoritmo,
 - le strutture dati su cui l'algoritmo opera.
- Codificare algoritmo e strutture dati in un linguaggio comprensibile dalla macchina virtuale utilizzata (in questo corso useremo il linguaggio C)

Un *algoritmo* è un insieme di regole operative (non ambigue) che lega i risultati voluti ai dati di ingresso.

Gli algoritmi possono essere descritti in vari modi; ad esempio, mediante un linguaggio pseudo-naturale, un diagramma di flusso, o un linguaggio di programmazione vero e proprio.

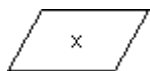
Diagrammi di Flusso

I diagrammi di flusso sono un linguaggio grafico per descrivere algoritmi.

I valori utilizzati per il calcolo sono contenuti in variabili. Possiamo paragonare una variabile ad un contenitore che contiene un valore che può essere letto e può essere rimpiazzato con un altro valore.

Il calcolo è fatto attraverso l'esecuzione di istruzioni. L'istruzione fondamentale è l'assegnamento di un valore ad una variabile. Nel calcolo dei valori, utilizziamo *espressioni* che sono formate con valori numerici (costanti), variabili, e operatori.

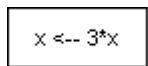
Un diagramma di flusso rappresenta le istruzioni entro blocchi e la sequenza di esecuzione delle istruzioni mediante l'utilizzo di frecce. Noi considereremo i seguenti tipi di blocco:



Ingresso/uscita : viene richiesto un valore e viene inserito nella variabile x

oppure

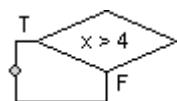
il valore contenuto nella variabile x viene visualizzato



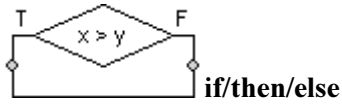
Attività: in questo caso alla variabile x viene assegnato il valore $3x$



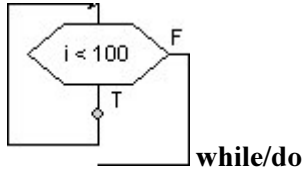
Chiamata di funzione



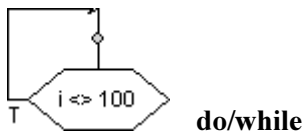
Scelta if/then Se $x > 4$ allora



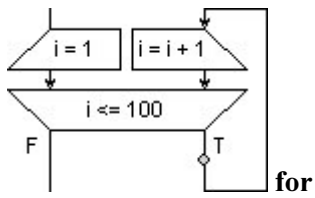
Se $x > y$ allora altrimenti



Ciclo while: mentre $i < 100$ esegui



Ciclo do/while: ripeti Quando $i <= 100$



Ciclo for: per I da 1 a 100, con incremento 1, esegui ...