

Lab. Programmazione (CdL Informatica)
&
Informatica (CdL Matematica)
a.a. 2022-23

Monica Nesi

Università degli Studi dell'Aquila

28 Settembre 2022

Diagrammi di flusso (Flow-chart)

Un metodo *grafico* per formalizzare algoritmi (soluzioni di problemi).

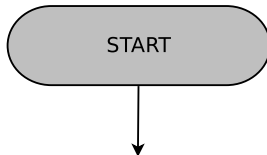
Un altro metodo per descrivere algoritmi è il *pseudo-linguaggio* tramite *pseudo-codice*.

Un *diagramma di flusso* è un metodo per descrivere un algoritmo in modo formale che:

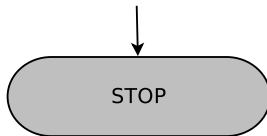
- ▶ utilizza *blocchi base* per rappresentare azioni (o istruzioni) e decisioni (solo *binarie*, i.e. una logica a 2 valori);
- ▶ ha *archi orientati* per descrivere la sequenza di svolgimento delle azioni (*flusso* dell'esecuzione).

Blocchi di inizio e fine

Inizio

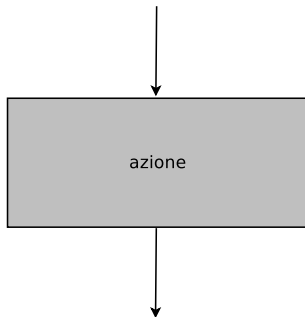


Fine



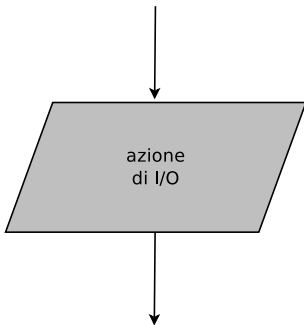
Blocco di azione

Azione o istruzione, i.e. operazione elementare della macchina



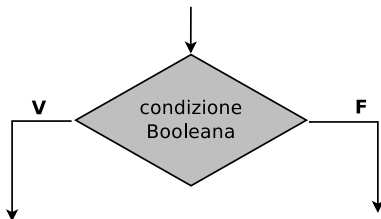
Blocco di Input/Output

Operazioni di acquisizione dati (e.g. lettura) e di restituzione dei risultati (e.g. scrittura, stampa, visualizzazione)



Blocco di decisione binaria

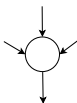
Sia data una condizione (o proprietà) logica o *booleana*, che ha valore *vero* o *falso*:¹



¹George Boole (1815-1864), matematico e logico britannico, considerato il fondatore della Logica Matematica.

Connettore

Consente di *riunire* possibili diversi flussi di esecuzione in uno solo.



Regole per un diagramma di flusso

In un diagramma di flusso occorre soddisfare le seguenti proprietà:

- ▶ uno ed un solo blocco di inizio;
- ▶ uno ed un solo blocco di fine;
- ▶ tutti gli archi devono avere origine e fine in un blocco.

Costrutti di controllo: sequenzializzazione

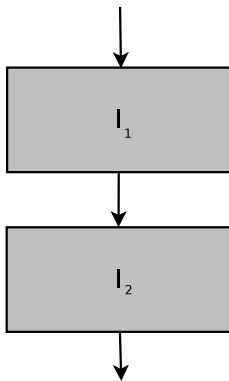
I *costrutti di controllo* permettono di esprimere il flusso dell'esecuzione.

Siano I , I_1 , I_2 istruzioni o azioni (op. elementari della macchina).

Sequenzializzazione:

per indicare che I_1 deve essere eseguita prima di I_2
(o equivalentemente che I_2 deve essere eseguita dopo I_1),
ovvero I_1 ed I_2 sono in sequenza,
si rappresenta prima il blocco di azione I_1 e poi il blocco di azione I_2 .

Sequenzializzazione



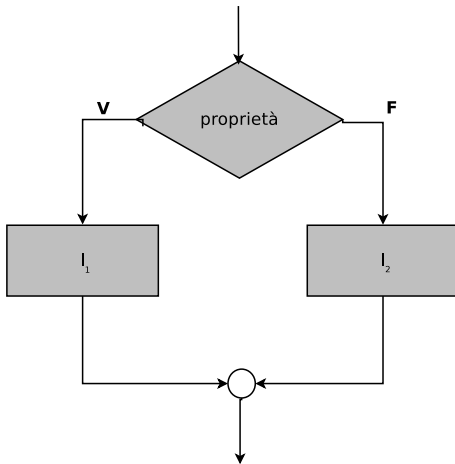
Costrutti di controllo: condizionale

Condizionale (*if-then-else*):

data una proprietà o condizione logica, per indicare che I_1 viene eseguita se la proprietà è vera ed I_2 viene eseguita se la proprietà è falsa, si combina il blocco di decisione binaria con i blocchi di azione per I_1 ed I_2 ed un connettore.

N.B.: I_1 ed I_2 sono *mutuamente esclusive* (o si esegue l'una o l'altra, mai entrambe).

Condizionale

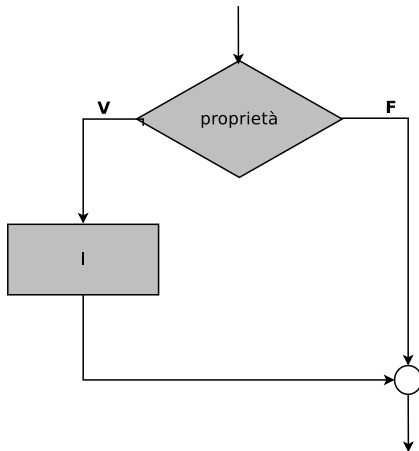


Costrutti di controllo: condizionale (senza else)

Condizionale (*if-then*):

data una proprietà o condizione logica, per indicare che l'azione *I* viene eseguita se la proprietà è vera e non si esegue alcuna azione se la proprietà è falsa, si usa la seguente rappresentazione grafica.

Condizionale (senza else)



Costrutti di controllo: iterazione o ciclo

Iterazione o Ciclo (*while*):

data una proprietà o condizione logica, per indicare che l'istruzione *I* deve essere ripetuta (*iterata*) fintantoché la proprietà è vera, si rappresenta un blocco di decisione binaria con un arco che torna a valutare la proprietà.

L'esecuzione di questo costrutto avviene come segue.

Per prima cosa si valuta la proprietà.

Se la proprietà è vera, si esegue *I* e poi si *torna in ciclo* a valutare nuovamente la proprietà.

Altrimenti (i.e. la proprietà è falsa) si *esce dal ciclo* andando avanti con l'istruzione che segue il costrutto iterativo.

Iterazione o ciclo

