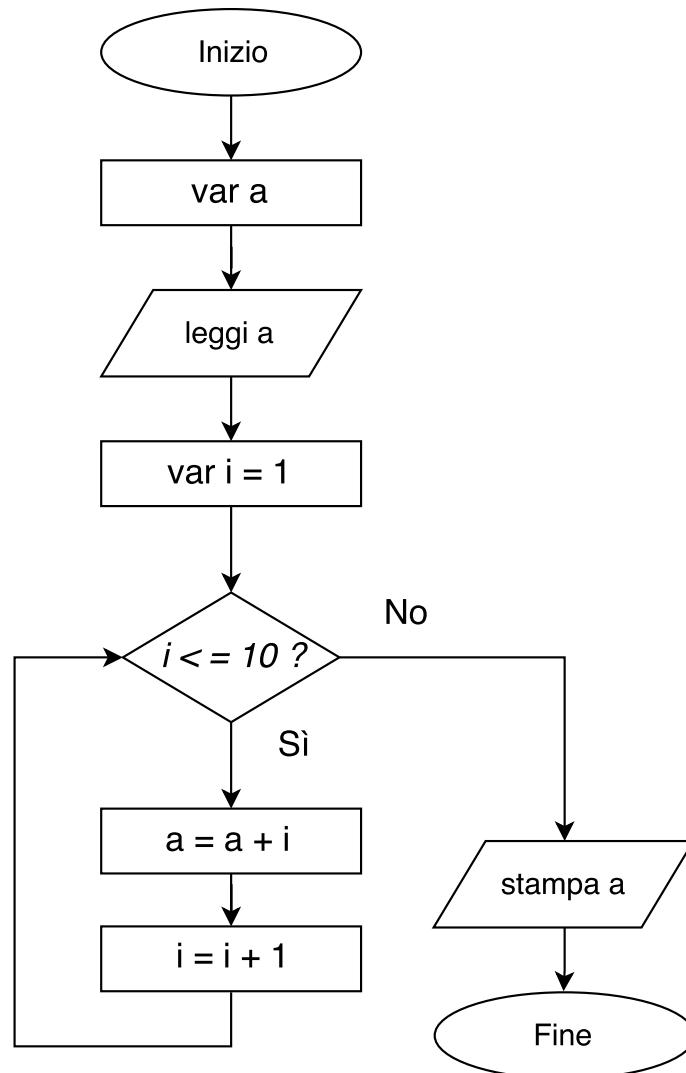


## Diagrammi di flusso - Flowcharts

E' un formalismo che consente di rappresentare graficamente gli algoritmi

- Un diagramma di flusso descrive le azioni da eseguire ed il loro ordine di esecuzione
- Ogni azione elementare corrisponde ad un simbolo grafico (blocco) diverso
- Ogni blocco ha un ramo in ingresso ed uno o più rami in uscita; collegando tra loro i vari blocchi attraverso i rami, si ottiene un diagramma di flusso
- Un diagramma di flusso appare, quindi, come un insieme di blocchi di forme diverse che contengono le istruzioni da eseguire, collegati fra loro da frecce che specificano la sequenza in cui i blocchi devono essere eseguiti (flusso del controllo di esecuzione).

# Diagrammi di flusso - Flowcharts



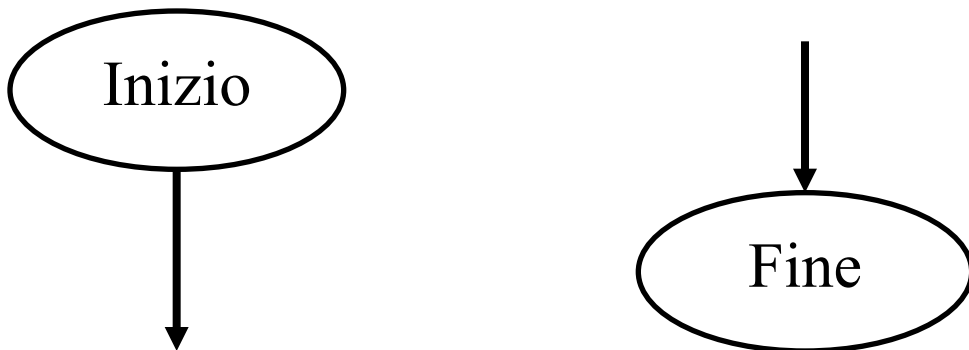
- L'insieme dei dati di ingresso e dei risultati vengono rappresentati attraverso dei nomi simbolici, detti **variabili** (ad esempio, «a»)
- Può essere inoltre necessario introdurre delle variabili "temporanee" (ad esempio, i), necessarie alla risoluzione del problema: tali variabili vengono anch'esse rappresentate da nomi simbolici.

## Diagrammi di flusso - Espressioni

- Sequenze di variabili e costanti combinate fra loro mediante operatori (ad es, operatori aritmetici: +, -, \*, /, %, sqrt);
- Nella valutazione di una espressione, si sostituisce ad ogni variabile il suo valore attuale e si eseguono le operazioni secondo un ordine prestabilito da regole di precedenza (possono comparire parentesi).
- A tutte le variabili che compaiono nell'espressione deve essere stato associato un valore prima della valutazione dell'espressione
- Espressioni relazionali e logiche: danno come risultato vero o falso (> minore, < maggiore, = uguale, != diverso)

## Inizio e fine esecuzione

- **Inizio** è il blocco da cui deve iniziare l'esecuzione dell'algoritmo (uno solo). La freccia è sempre uscente dal blocco inizio.
- **Fine** è il blocco che fa terminare l'esecuzione dell'algoritmo (almeno uno). La freccia è sempre entrante nel blocco fine

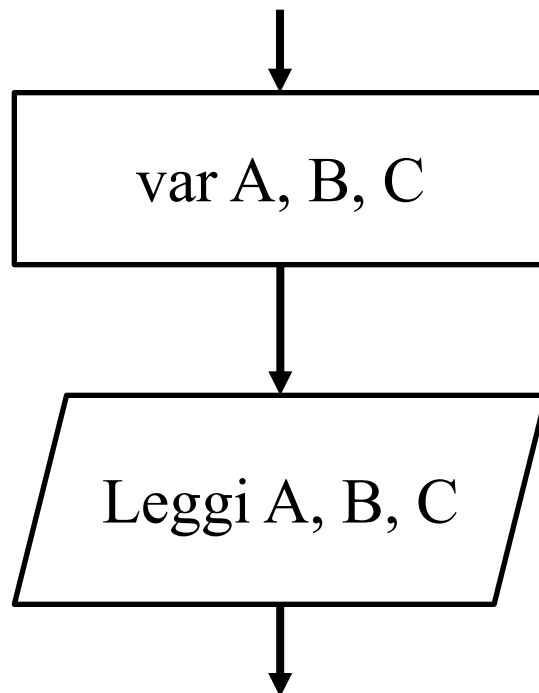


- A questi blocchi non corrisponde alcuna azione!

### Ingresso (lettura, read, input)

- **Esecuzione dell'istruzione:**

Si ricevono dall'unità di ingresso (per esempio, la tastiera) tanti valori quante sono le variabili specificate all'interno del blocco

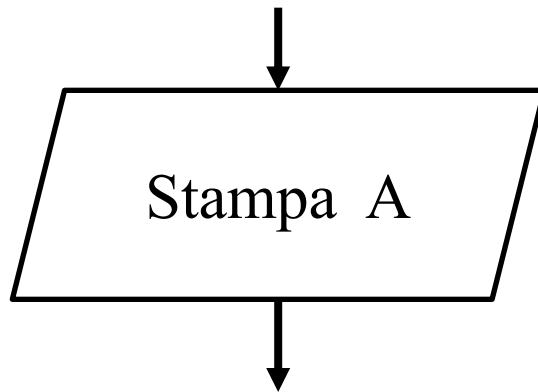


- «Crea le variabili A,B,C. Leggi i tre valori da tastiera, e salvali rispettivamente nelle variabili A, B e C»

### Uscita (stampa, print, output)

- **Esecuzione dell'istruzione:**

Si calcolano i valori delle espressioni e si trasmettono all'unità di uscita (ad esempio, il video)

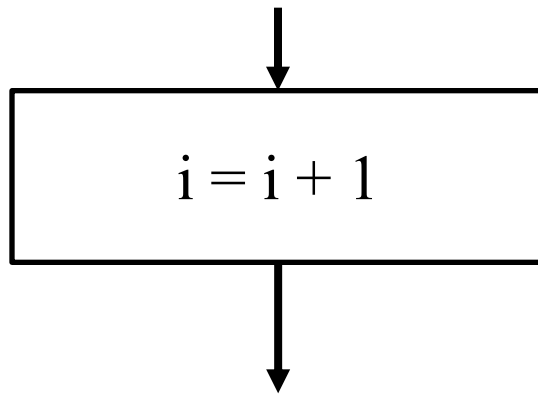


- «calcola il valore dell' espressione A e stampala a video»
- N.B. il valore di A non viene alterato dall'esecuzione del blocco

### Assegnamento

- **Esecuzione dell'istruzione:**

Si calcola il valore dell'espressione a destra del simbolo «=» e lo si salva (*assegna*) nella variabile indicata a sinistra del simbolo «=»

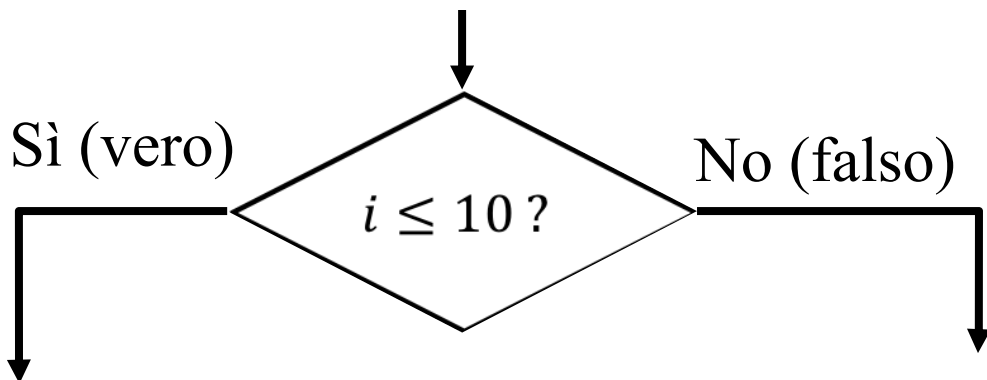


- «Calcola il valore dell'espressione  $i+1$  ed assegnalo alla variabile  $i$  »
- Esempio: se  $i$  vale 6 prima di entrare nel blocco, calcolo  $6+1$  (7) e lo salvo nella variabile  $i$ .
- Quando si esce dal blocco la variabile  $i$  vale 7.

## Condizioni

- **Esecuzione dell'istruzione:**

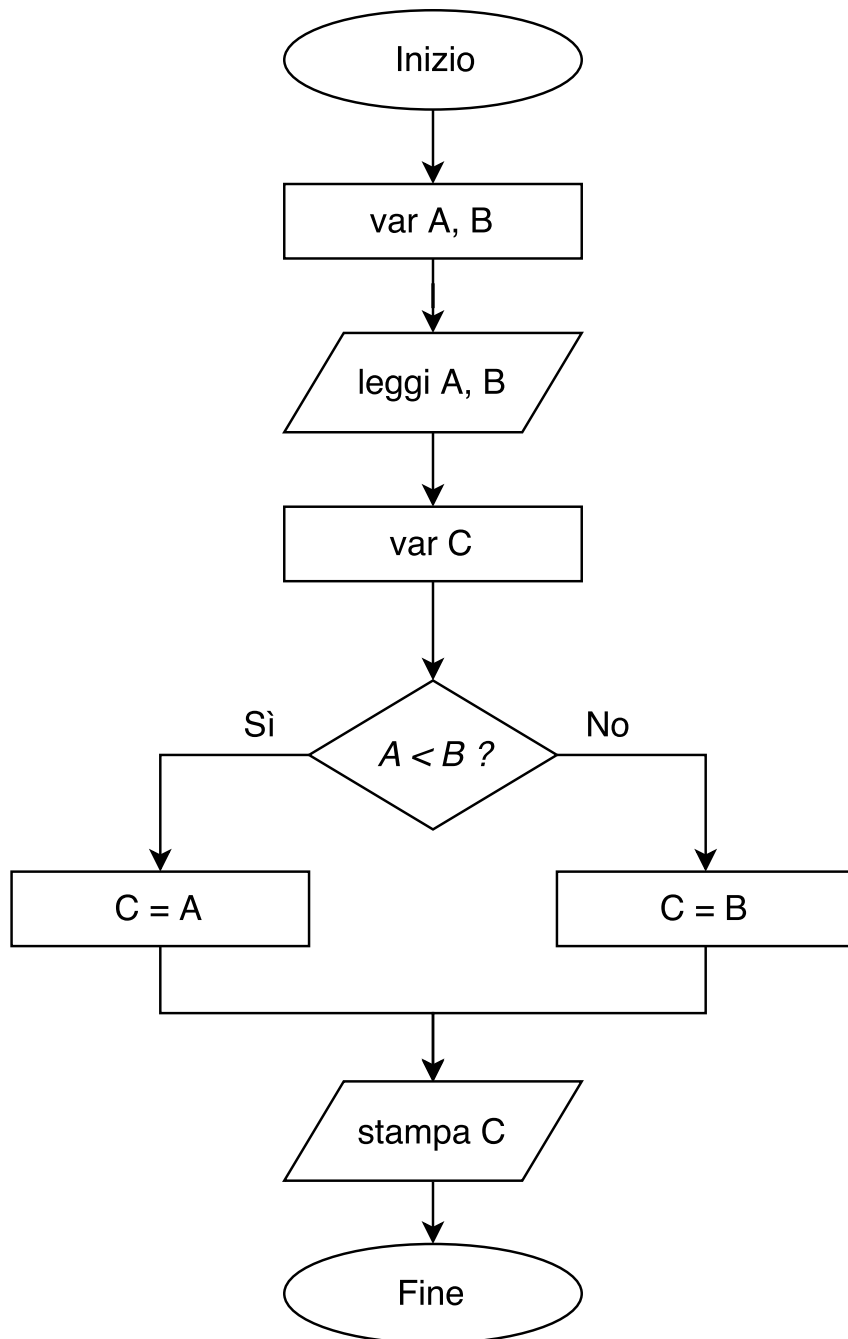
Si valuta la condizione specificata all'interno del blocco: se è verificata, si prosegue con la line di flusso contrassegnata da «Sì», altrimenti (se non è verificata) si prosegue con il ramo etichettato con «No»



- «Calcola il valore dell'espressione  $i \leq 10$ : se è vero, prosegui per il ramo Sì, altrimenti prosegui per il ramo No»
- Il blocco condizione è l'elemento base per realizzare **alternative** e **ripetizioni**.



## Esempio



- *Cosa fa questo algoritmo?*

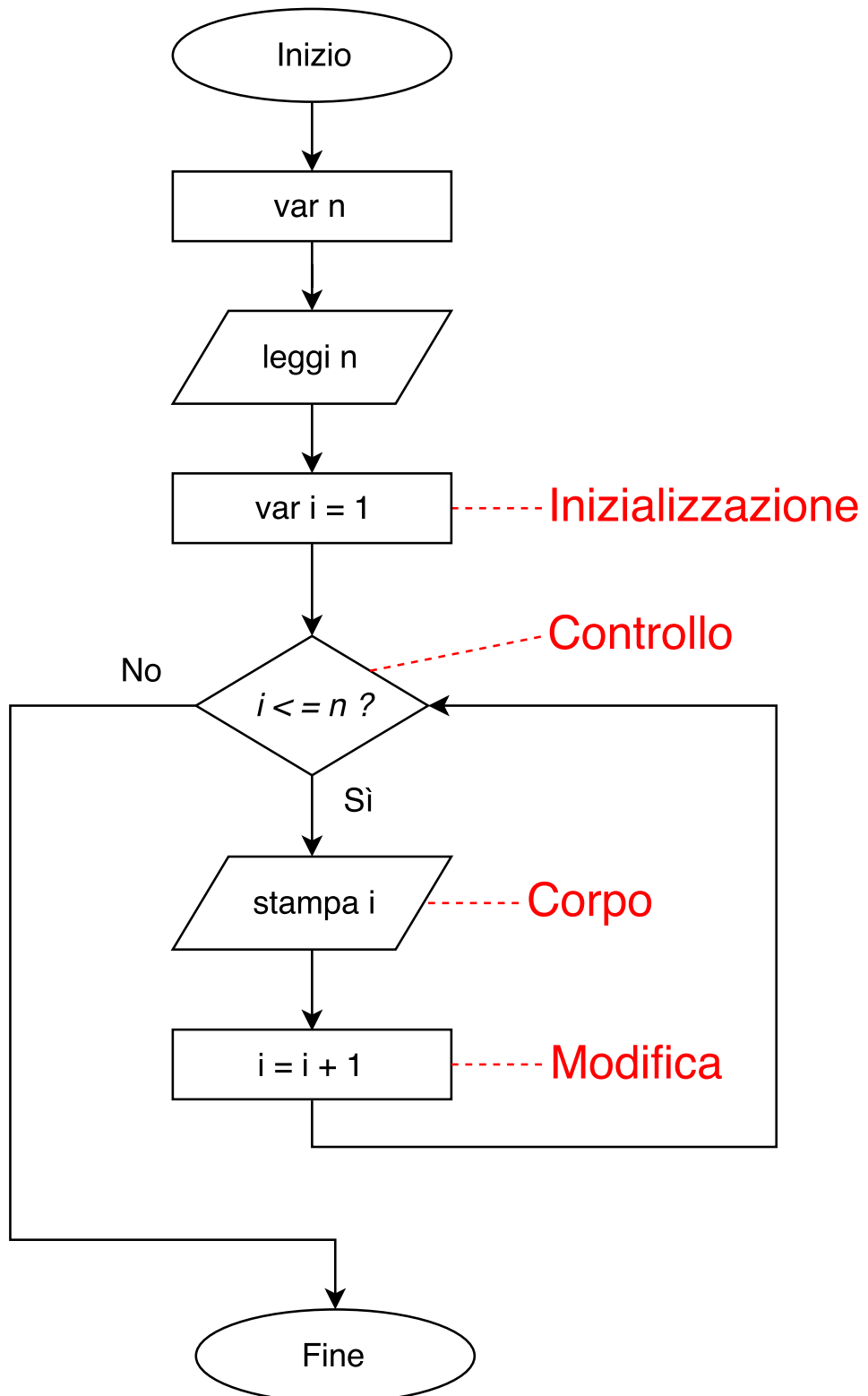
### Iterazione o Ripetizione

Esprime la ripetizione di una sequenza di istruzioni.

Nel caso più generale (ripetizione enumerativa), è costituita da:

- **Inizializzazione:** assegnazione dei valori iniziali alle variabili caratteristiche del ciclo (viene eseguita una sola volta);
- **Controllo:** determina, in base al valore delle variabili che controllano l'esecuzione del ciclo se il ciclo deve essere ripetuto o meno;
- **Corpo:** esecuzione delle istruzioni fondamentali del ciclo che devono essere eseguite in modo ripetitivo;
- **Modifica:** modifica dei valori che controllano l'esecuzione del ciclo (eseguito ad ogni iterazione).

## Iterazione



In generale, qualsiasi algoritmo si compone attraverso le seguenti strutture:

- **Concatenazione:** istruzioni di lettura, scrittura o assegnamento eseguite in sequenza
- **Alternative:** mediante blocchi condizionali
- **Iterazioni**

Chiaramente, la complessità di un algoritmo potrà essere alta o bassa, ma sarà sempre e in ogni caso esprimibile attraverso queste strutture base (composte in modo vario, come componiamo pezzi di lego), che formano la cosiddetta base della programmazione “strutturata”.

## Possibili esercizi da fare

Mediante diagrammi di flusso, descrivere le operazioni da eseguire per poter completare la richiesta:

- Leggere un numero da tastiera e stampare a video se è positivo o negativo
- Leggere da tastiera due numeri e stampare a video la loro somma
- Leggere da tastiera due numeri e stampare a video il maggiore dei due
- Calcolare il risultato dell'equazione di primo grado leggendo da tastiera i coefficienti dell'equazione  $ax+b = 0$  e stampando il valore di  $x$
- Leggere tre numeri da tastiera e stampare a video il massimo tra i tre numeri