

# L'istruzione di selezione doppia

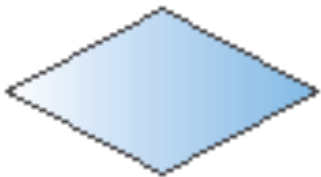
---

- Nella descrizione di un algoritmo generalmente che le operazioni da eseguire sono diverse a seconda dei dati inseriti
- Se ad esempio dobbiamo comperare un oggetto, dobbiamo verificare se i soldi a disposizione sono sufficienti e si prospettano due soluzioni alternative:
  - se** i soldi ci bastano
  - allora** compriamo l'oggetto
  - altrimenti** se i soldi non sono sufficienti, dobbiamo procurarne degli altri

# L'istruzione di selezione doppia

---

- Nel diagramma a blocchi, questa situazione è illustrata ricorrendo a un nuovo elemento grafico, detto blocco di **test** o di **blocco di confronto**



- al suo interno vengono effettuate operazioni di confronto e il loro risultato può essere soltanto **SÌ** o **NO** (**VERO** o **FALSO**)

# L'istruzione di selezione doppia

---

- Esempi di istruzioni di confronto (o **test**) che possono essere scritte nei blocchi condizionali:

*Ho soldi sufficienti per comperare la moto?*

*Il numero 50 è maggiore di 0?*

*Mario è un alunno di questa classe?*

*Il serbatoio è pieno?*

*Una mucca depone le uova?*

*27 è un numero pari?*

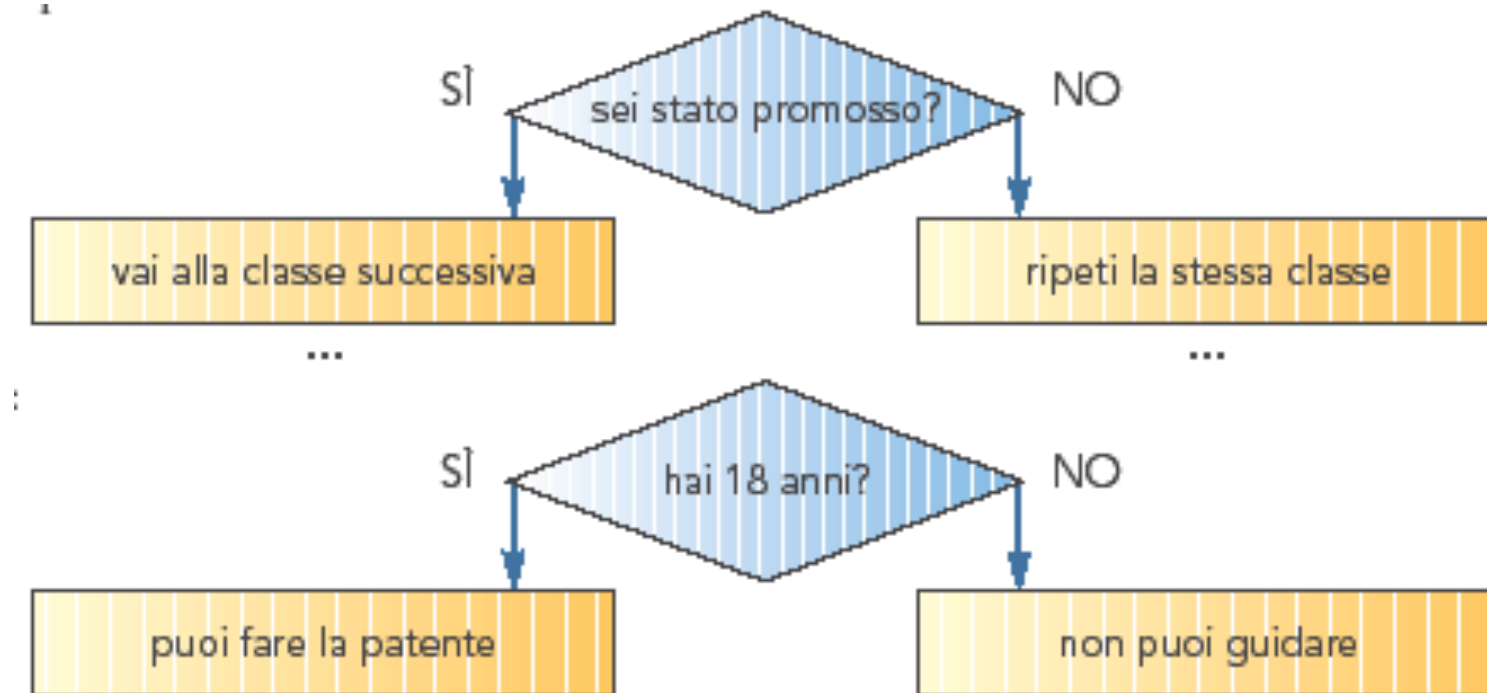
# L'istruzione di selezione doppia

---

- L' unica possibile risposta a queste domande è **SÌ** oppure **NO**
- Esistono soltanto due alternative: si tratta dunque di “istruzioni di confronto”
- Prendono il nome di **condizioni logiche**
- A seconda del valore della risposta, è sempre necessario intraprendere percorsi alternativi

# L'istruzione di selezione doppia

- Vediamo un esempio:



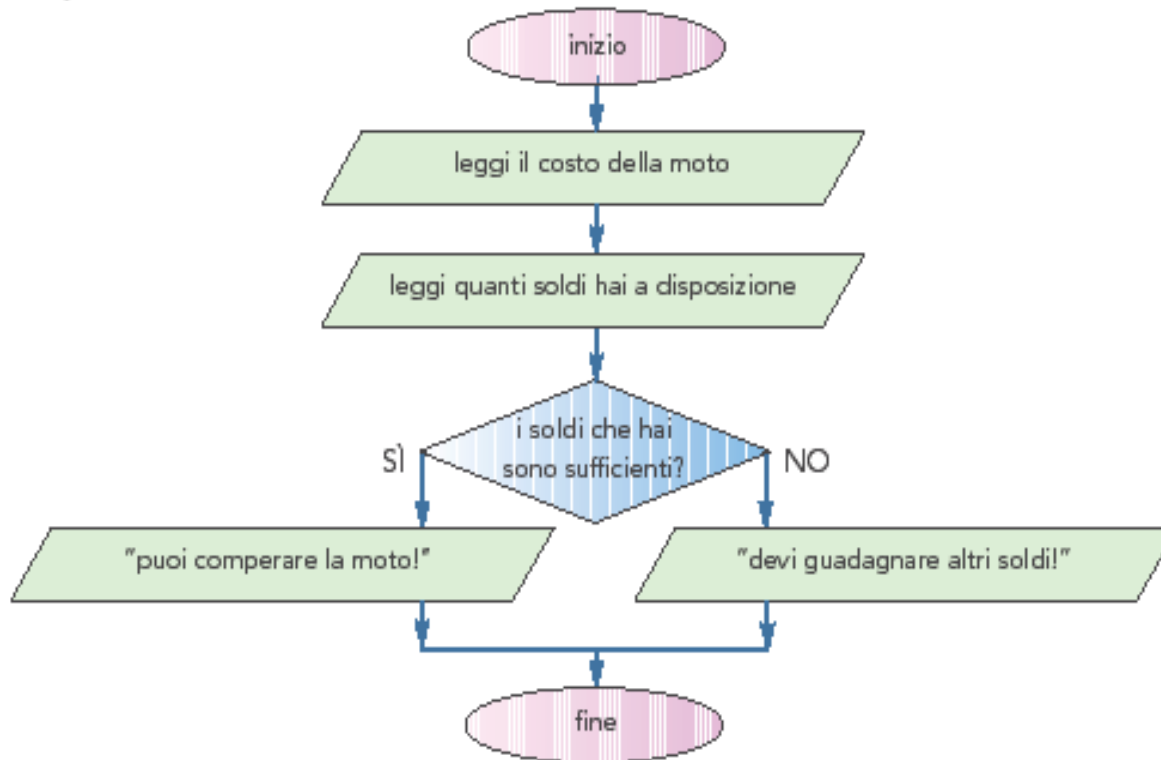
# L'istruzione di selezione doppia

---

- A seconda della risposta data alla domanda (cioè al **test**), le strade “si dividono” e vengono eseguite operazioni diverse nei due “rami”
- Poiché siamo in presenza di istruzioni in entrambi i rami del diagramma, parliamo in questo caso di **selezione doppia**

# L'istruzione di selezione doppia

- Il seguente diagramma descrive la procedura per l'acquisto di una moto:



# L'istruzione di selezione doppia

---

- Traduciamo questa istruzione in **linguaggio di progetto**
- Dobbiamo scrivere una istruzione per riga: è necessario stabilire una regola che ci permetta di riconoscere in modo chiaro **l'istruzione di selezione**
- Utilizziamo una **scrittura incolonnata** in modo particolare, ricorrendo a tre parole che ci consentono di individuare l'istruzione di **test** e i due percorsi alternativi



# L'istruzione di selezione doppia

---

**se** *<il test da' risultato vero>*

**allora**

*<esegui questa istruzione>*

**altrimenti**

*<esegui quest'altra istruzione>*

# L'istruzione di selezione doppia

---

- Il programma completo è :

**inizio**

**leggi** quanto costa la moto

**leggi** quanti soldi hai a disposizione

**se** i soldi che hai sono sufficienti

**allora**

**scrivi** “puoi comperare la moto!”

**altrimenti**

**scrivi** “devi guadagnare altri soldi”

**fine**

# L'istruzione di selezione doppia

---

- L' **incolonnamento** utilizzato consente di riconoscere subito le istruzioni interne ad una istruzione di selezione
- Infatti sono spostate più a destra di quest' ultima
- le due operazioni di scrittura vengono scritte **alternativamente**, a seconda del risultato del test

# L'istruzione di selezione doppia

---

- Con la notazione che abbiamo adottato risulta semplice seguire il flusso del programma nel linguaggio di progetto

```
se <condizione>  
  allora  
    esegui il primo ramo  
  altrimenti  
    esegui il secondo ramo
```

# L'istruzione di selezione doppia

---

- Le **regole utilizzate** per scrivere il programma in modo incolonnato prendono il nome di **regole di indentazione**
- Non sono uniche né obbligatorie, e ogni programmatore può scegliere di scrivere un programma nel modo che più lo facilita
- Unico obbligo da rispettare è la “**uniformità**”: una volta stabilite devono essere applicate sempre nello stesso modo

# L'istruzione di selezione doppia

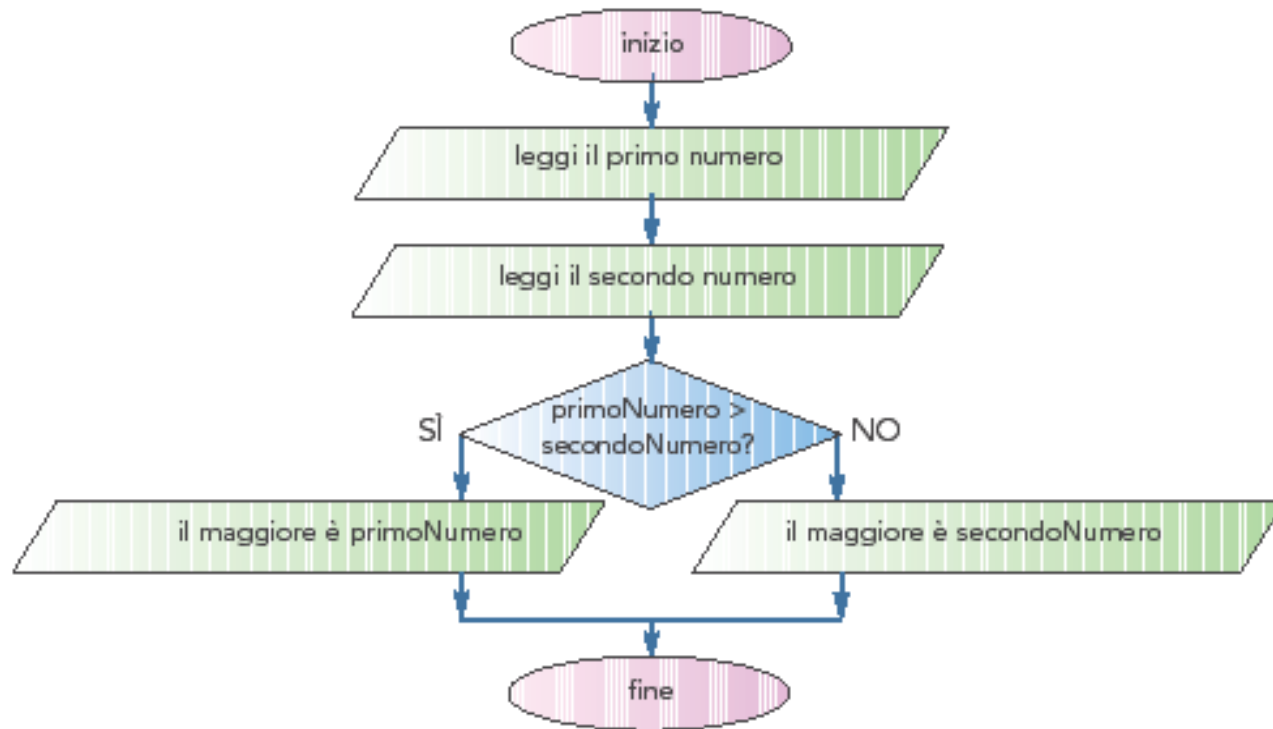
---

- Esempio 2 **Determinare il maggiore tra due numeri**
- Leggiamo due numeri da tastiera e determiniamo quale dei due è il maggiore
- All'interno della istruzione di test è necessario effettuare il confronto tra i due numeri, quindi l'istruzione condizionale è la seguente:

**primoNumero > secondoNumero**

# L'istruzione di selezione doppia

- Il diagramma di flusso è il seguente:



# L'istruzione di selezione doppia

---

- Traduciamo l'algoritmo in linguaggio di progetto

**inizio**

**leggi** il primo numero

**leggi** il secondo numero

**se** il primo numero è  $>$  del secondo numero

**allora**

**scrivi** "il maggiore è il primo numero"

**altrimenti**

**scrivi** "il maggiore è il secondo numero"

**fine**



# Le selezione semplice

---

- E' possibile che le operazioni da eseguire siano presenti in un solo ramo
- Ad esempio quando si deve effettuare una operazione in una sola situazione
- In questo caso si parla di **selezione semplice**

# Le selezione semplice

---

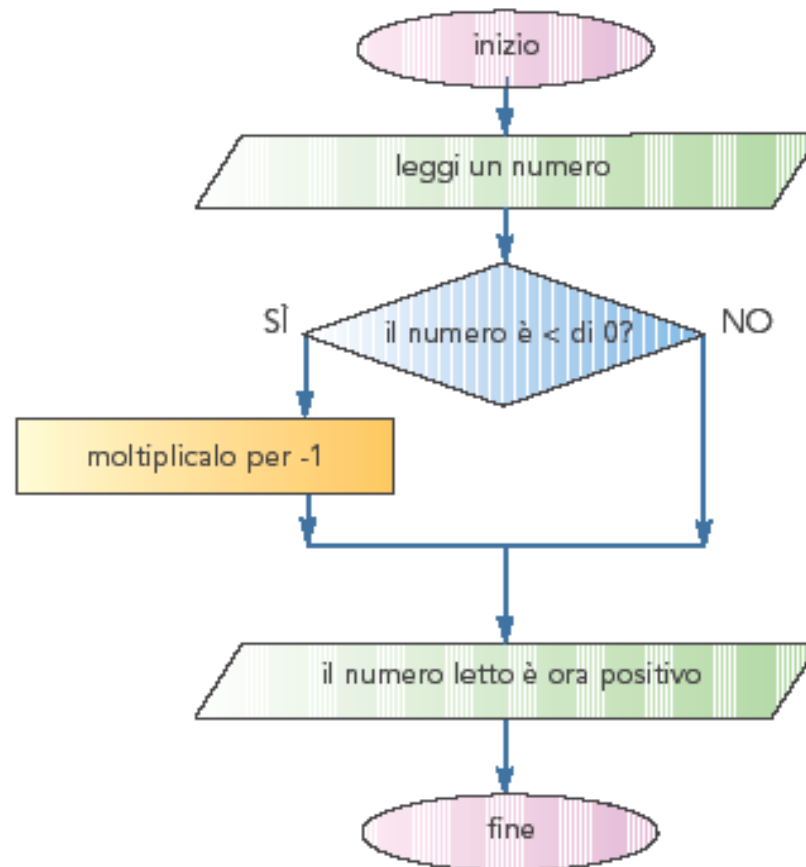
- esempio 3 **Trasformazione di numeri negativi in positivi**

Scriviamo un programma che trasforma i numeri negativi in numeri positivi:

- se il numero è già positivo non dobbiamo effettuare nessuna operazione
- se è negativo lo moltiplichiamo per  $-1$ , in modo da cambiarne il segno

# Le selezione semplice

---



# L'istruzione di selezione doppia

---

- In linguaggio di progetto è il seguente:

**inizio**

**leggi** un numero

**se** il numero è  $< 0$

**allora**

**esegui** moltiplicalo per  $-1$  in modo da renderlo positivo

**scrivi** “ora il numero letto è sicuramente positivo”

**fine**