

# RAPPRESENTAZIONE DEGLI ALGORITMI

---

Fondamenti di Programmazione 2022/2023



# Come si esprime un algoritmo?

---

- Per realizzare un procedimento risolutivo, un algoritmo esegue delle **operazioni** su un **insieme di informazioni**
- Per definire correttamente un algoritmo è quindi necessario specificare in maniera formale (precisa e non ambigua):
  - le **informazioni** su cui l'algoritmo lavora
  - le **operazioni** che l'algoritmo compie



# Organizzazione dell'informazione in un algoritmo

---

- All'interno di un algoritmo un'informazione può essere organizzata in vari modi:
  - **variabile**
  - **costante**
  - **espressione**



# Variabile

---

- Una variabile è un ente, appartenente ad un certo tipo, che **può assumere uno qualunque dei valori appartenenti al tipo**.
- **Una variabile è identificata da un nome**, che riflette il ruolo che questa assume all'interno dell'algoritmo.
- Il valore di una variabile può essere sia **utilizzato** (lettura) che **modificato** (scrittura).



# Variabile

---

- Nell'algoritmo per il calcolo del MCD i valori di X, Y e R sono ospitati in altrettante variabili

1. Leggi due numeri  $X$  e  $Y$ , con  $X > Y$
2. Dividi  $X$  per  $Y$  e ottieni il resto  $R$
3. Se  $R=0$ , termina: il MCD è  $Y$
4. Sostituisci  $X$  con  $Y$
5. Sostituisci  $Y$  con  $R$
6. Torna al punto 2.



# Costante

---

- E' un oggetto, appartenente ad un certo tipo, **il cui valore rimane immutato durante l'esecuzione dell'algoritmo.** Ad una costante può essere attribuito un nome.
- Esempi:
  - 0            è una costante di tipo intero
  - 3.1415    è una costante di tipo reale
  - pigreco   è una costante di tipo reale e valore 3.1415



# Costante

---

- Nell'algoritmo per il calcolo del MCD è presente una sola costante

1. Leggi due numeri  $X$  e  $Y$ , con  $X > Y$
2. Dividi  $X$  per  $Y$  e ottieni il resto  $R$
3. Se  $R=0$ , termina: il MCD è  $Y$
4. Sostituisci  $X$  con  $Y$
5. Sostituisci  $Y$  con  $R$
6. Torna al punto 2.



# Espressione

---

- E' una **sequenza** di operandi, operatori e parentesi, dove gli operandi possono essere variabili o costanti. Il tipo dell'espressione complessiva dipende dai tipi degli operandi coinvolti nell'espressione.
- Esempi ( $a, b$  variabili intere;  $x, y$  variabili reali)
  - $a*b+50$  è un'espressione di tipo intero
  - $a*3.1415$  è un'espressione di tipo reale
  - $x/2$  è un'espressione di tipo reale
  - $2*b*\text{pigreco}$  è un'espressione di tipo reale





# Espressione

---

- Non ci sono espressioni?

1. Leggi due numeri  $X$  e  $Y$ , con  $X > Y$
2. Dividi  $X$  per  $Y$  e ottieni il resto  $R$
3. Se  $R=0$ , termina: il MCD è  $Y$
4. Sostituisci  $X$  con  $Y$
5. Sostituisci  $Y$  con  $R$
6. Torna al punto 2.



# Espressione

---

- Non ci sono espressioni?

```
1. Leggi due numeri X e Y, con  $X > Y$   
2.  $R \leftarrow X - [X/Y] * Y$   
3. Se  $R=0$ , termina: il MCD è Y  
4.  $X \leftarrow Y$   
5.  $Y \leftarrow R$   
6. Torna al punto 2.
```



# Rappresentazione delle operazioni

- Come definire e rappresentare in maniera non ambigua le istruzioni che costituiscono l'algoritmo?
- Due possibili strumenti:
  - **Pseudo codice**
  - **Diagrammi di flusso (Flow Chart)**

**Algorithm E** (*Euclid's algorithm*). Given two positive integers  $m$  and  $n$ , find their *greatest common divisor*, that is, the largest positive integer that evenly divides both  $m$  and  $n$ .

**E1.** [Find remainder.] Divide  $m$  by  $n$  and let  $r$  be the remainder. (We will have  $0 \leq r < n$ .)

**E2.** [Is it zero?] If  $r = 0$ , the algorithm terminates;  $n$  is the answer.

**E3.** [Reduce.] Set  $m \leftarrow n$ ,  $n \leftarrow r$ , and go back to step E1. ■

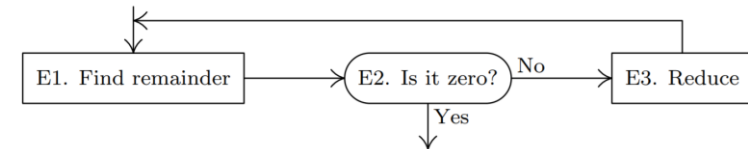


Fig. 1. Flow chart for Algorithm E.



# ACHTUNG!!! Linguaggio naturale

---

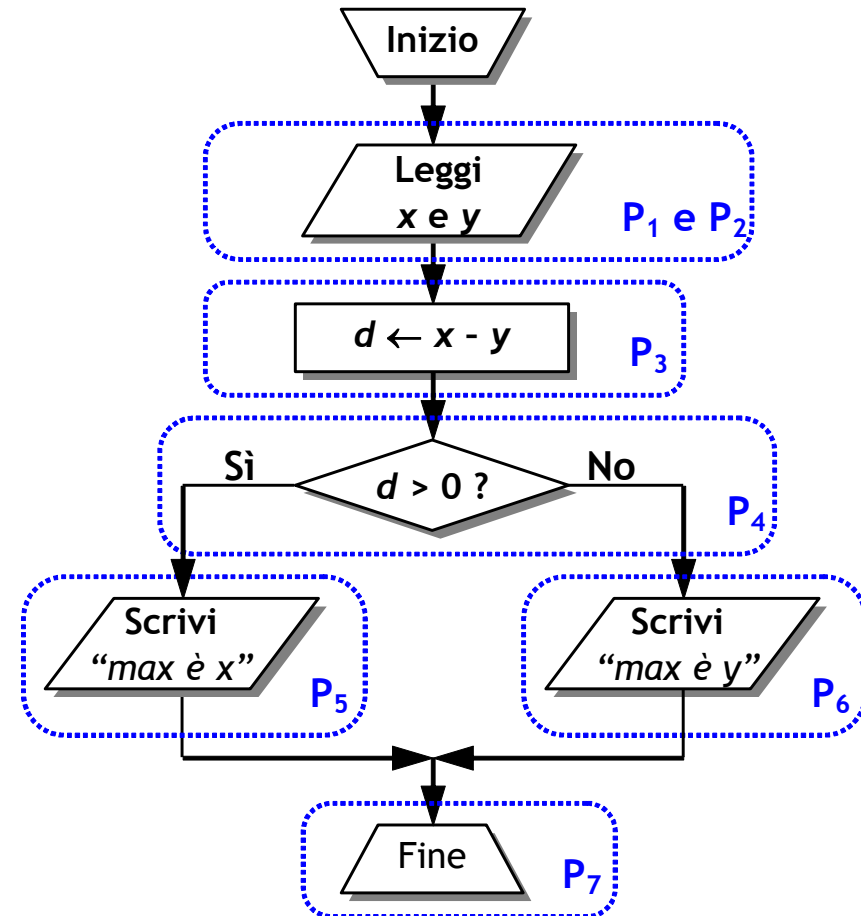
- Nella risoluzione di un problema, il primo passo verso la modellazione di un algoritmo è cercare di descrivere le azioni da fare in linguaggio naturale
- Es. Trovare il massimo tra due numeri
- Passi:
  - Prendo il primo numero
  - Prendo il secondo numero
  - Faccio la differenza
  - Se il risultato è maggiore di 0 allora il primo numero è il massimo
  - ...

**Preciso?**  
**Non ambiguo?**



# Diagramma di flusso (flow chart)

- Consente la modellazione grafica di un algoritmo
- Alternativa allo pseudo-codice per algoritmi non eccessivamente complessi
- Molto immediato: consente di descrivere un algoritmo concentrandosi principalmente sulla sequenza delle operazioni di cui si compone.



# Pseudo codice

---

- Rappresentazione dell'algoritmo in forma testuale.
- Costrutti di controllo spesso descritti con forme e parole chiave corrispondenti o vicine a quelle dei linguaggi di programmazione
- Le particolari operazioni possono essere descritte in modo informale e sintetico.

```
leggi x,y  
d:= x-y  
if (d > 0)  
    stampa ("Max è X")  
else  
    stampa ("Max è y")
```



# Diagrammi di flusso

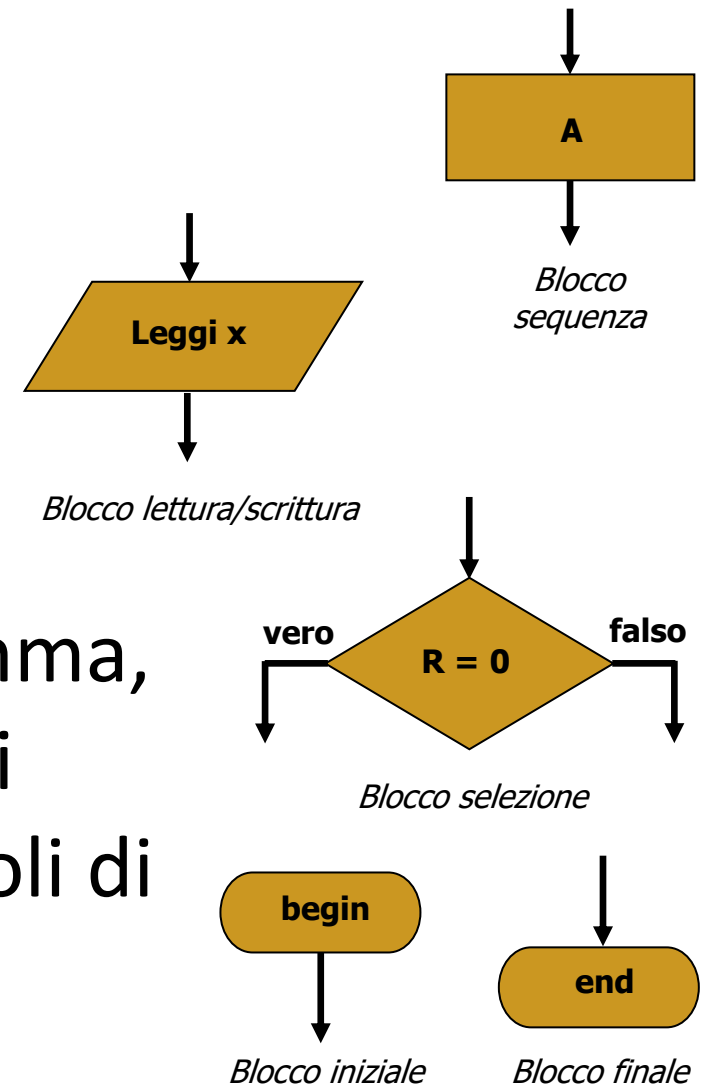
---

- Useremo il flow chart come rappresentazione grafica del flusso di controllo
- descrive il flusso delle operazioni da eseguire per realizzare il procedimento risolutivo definito dall'algoritmo, dai dati iniziali ai risultati
- ogni istruzione dell'algoritmo viene rappresentata all'interno di un blocco elementare, la cui forma grafica è determinata dal tipo di istruzione
- i blocchi sono collegati tra loro da linee di flusso, munite di frecce, che indicano il susseguirsi di azioni elementari



# Blocchi elementari del diagramma di flusso

- **Rettangolo**: indica una azione elaborativa
- **Parallelogramma**: indica un'operazione di input/output
- **Rombo**: indica una selezione
- **Ovale**: indica l'inizio o la fine di un programma, o di una sezione di codice (in questo caso si usano anche simboli cerchietto, detti simboli di connessione)





# Prime semplici operazioni

---

- Introduciamo le prime semplici operazioni che coinvolgono variabili, costanti ed espressioni
- In particolare:
  - Operazioni di calcolo e assegnazione
  - Operazioni di Ingresso/Uscita



# Costrutti di calcolo e assegnazione

---

- L'effetto è di aggiornare il valore di una variabile di un certo tipo con il valore ottenuto dalla valutazione di un'espressione dello stesso tipo.
- Il formato è:  
$$\text{variabile} = \text{espressione}$$
- **Achtung:** il segno  $=$  sta ad indicare l'**azione** di assegnare il valore a destra alla variabile il cui nome è a sinistra.  
Non è il segno dell'equazione come accade in  $a^2 + b^2 = c^2$



# Costrutti di calcolo e assegnazione

---

- Infatti a volte si usa il formato:

**variabile ← espressione**

per evitare confusioni (noi comunque useremo =).

- Esempi:

**a=4      a=a+1**

**b=0      a=a+b**

**b=a**

**cond= a > b**

**cond= (a>=0) and (a<=9)**



# Operazioni di calcolo e assegnazione

---

- Assumiamo che `i`, `j`, `val_m` siano variabili di tipo intero e che `cost_i` sia una costante sempre di tipo intero

- Quali sono le operazioni corrette?

`2040 = val_m`

`i + j = val_m`

`ci = val_m`

`i = j`



# Operazioni di ingresso/uscita

---

- Con le operazioni di input, il valore di una variabile viene modificato con il valore ottenuto grazie ad un'operazione di lettura dall'unità di ingresso (tastiera).
- Con le operazioni di output, un'espressione viene valutata ed il valore ottenuto viene presentato sull'unità di uscita (schermo).

