

CONCETTI FONDAMENTALI

Algoritmo	<i>Procedura</i> di trasformazione di un insieme di <i>dati iniziali</i> in un insieme di <i>risultati finali</i> mediante una sequenza di istruzioni.
Linguaggio di programmazione	<i>Linguaggio</i> (insieme di simboli e regole) <i>per rappresentare</i> le istruzioni di un algoritmo e la loro <i>concatenazione</i>
Programma	<i>algoritmo scritto in un linguaggio di programmazione al fine di comunicare al calcolatore elettronico le azioni da eseguire</i>
Processo	programma in <i>esecuzione</i>

ESEMPIO DI ALGORITMO

Uso di un distributore automatico di caramelle: costo £ 850

DESCRIZIONE PER L'ESECUTORE UMANO

1. inserire 9 monete da £ 100
2. tirare il cassetto corrispondente al tipo di caramelle desiderato
3. estrarre il pacchetto
4. richiudere il cassetto
5. ritirare il resto di £ 50

Prerequisito per l'esecutore umano:

bagaglio di nozioni comuni che permettono di *interpretare* le istruzioni e risolvere eventuali *ambiguità ed imprevisti*

REQUISITI DI UN ALGORITMO PER IL CALCOLATORE

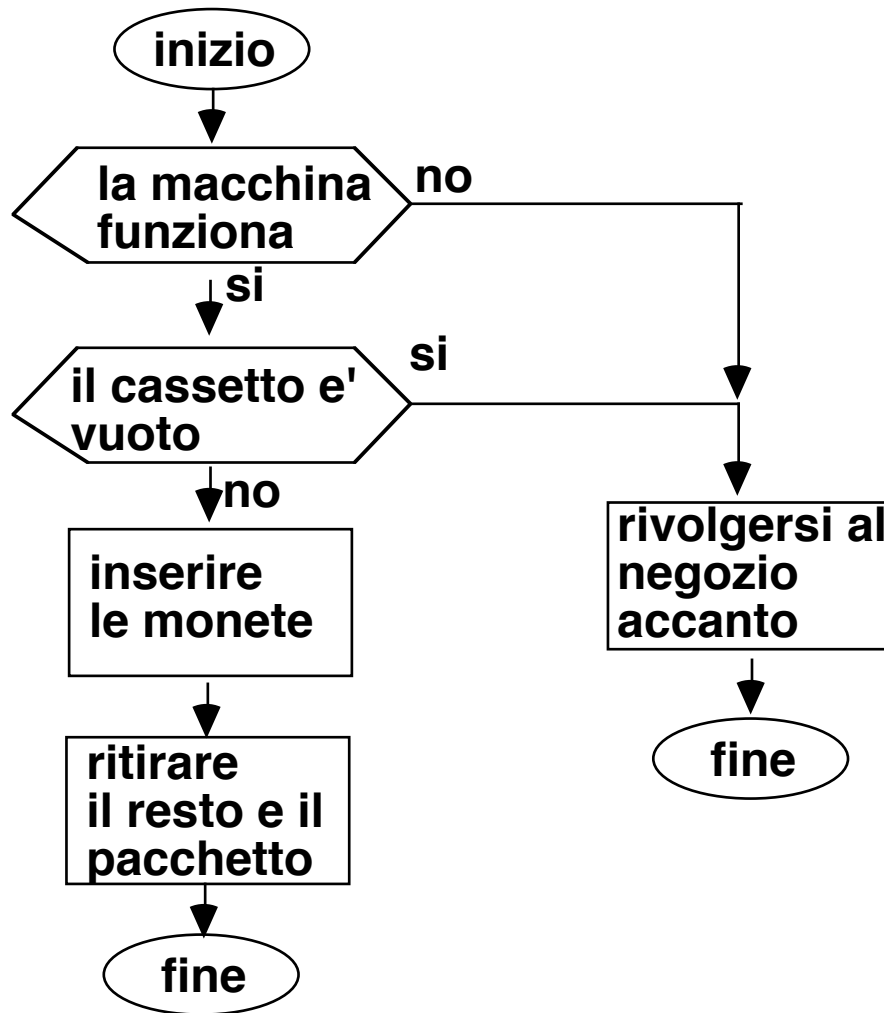
- ❑ istruzioni non ambigue
- ❑ istruzioni eseguibili in un tempo finito
- ❑ terminazione dell'esecuzione dopo un numero finito di passi

Descrizione intuitiva di un algoritmo

un formalismo grafico: i *diagrammi a blocchi* (*flow chart*).

- rappresentano la sequenza di istruzioni con linee di connessione orientate
- simboli grafici diversi per rappresentare tipi diversi di istruzioni
 - un esagono (o un rombo) : un *blocco di controllo* (condizione o *test*);
 - un rettangolo : un *blocco azione* (cioè un'istruzione imperativa);
 - il *blocco iniziale* e il *blocco finale* sono rappresentati con ellissi.

DESCRIZIONE CON DIAGRAMMA A BLOCCHI



- ❑ riduzione di ambiguità/incertezze tramite l'introduzione di test
- ❑ i test *decidono* quale è la sequenza delle istruzioni successive
- ❑ i test provocano una *bi-dimensionalità*
- ❑ problema delle azioni ripetute

Algoritmo per disegnare un poligono di N lati, ciascuno di lunghezza L



- ❑ I Linguaggi di programmazione di alto livello contengono strutture per poter esprimere la *ripetizione* delle istruzioni

```
INIZIO
  Leggi NLati
  Leggi Lato
  Ripeti Nlati volte
    Avanza Lato
    RuotaAdestra 360/ Nlati
FINE
```

TRADUZIONE DI UN PROGRAMMA IN LINGUAGGIO BINARIO

- ❑ Tutte le informazioni (programmi e dati) devono essere codificate in **bit**
bit (**b**inary **d**igit) ❑ valori <0> e <1> (dispositivi *binari*)
- ❑ Una sequenza di 8 bit è detto **byte**

PL1 - programma in linguaggio di alto livello

❑

❑ programma **traduttore**

PL2 - programma in linguaggio binario
comprensibile al calcolatore

Compilazione:

- Un programma, chiamato **compilatore**, prende in ingresso il programma PL1 (codice sorgente) e lo trasforma nel programma PL2 (codice binario) direttamente eseguibile sullo specifico calcolatore

Intepretazione:

- Un programma, chiamato **interprete**, prende in ingresso il programma PL1 e lo esegue “al volo” man mano che le istruzioni di PL1 sono lette, ovvero per ogni istruzione di PL1 fa eseguire al calcolatore le corrispondenti istruzioni

NEL NOSTRO CORSO DI FONDAMENTI DI INFORMATICA

Programma in linguaggio di alto livello

LINGUAGGIO LOGO



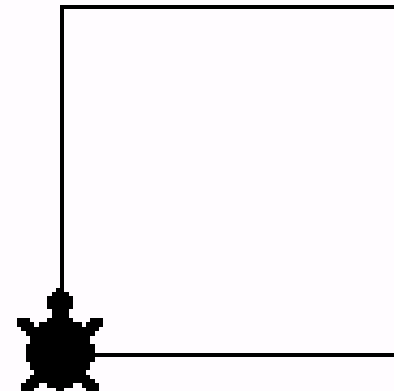
Programma in linguaggio binario
per calcolatore IBM/WINDOWS

□ programma interprete
MICROMONDI

Programma LOGO :

RIPETI 4 [AVANTI LATO
DESTRA 360 / NLATI]

MICROMONDI :



MODALITÀ D'ESAME

L'esame consiste in una **prova scritta** formata da due parti:

La **prima parte**, da realizzare **usando MICROMONDI**, consiste in esercizi del tipo

“Data una figura scrivere il programma LOGO per ottenere tale figura”

La **seconda parte**, da realizzare **senza usare MICROMONDI**, consiste in esercizi del tipo

“Data un programma LOGO disegnare la figura che si ottiene”