

Laboratorio di Informatica

Flusso del controllo: selezione

Laboratorio di Informatica

Flusso di controllo
Pagina 1

Teorema di Bohm e Jacopini

Tutti i programmi possono essere scritti in termini di tre strutture di controllo:

- **Sequenza**: permette di eseguire le istruzioni secondo l'ordine in cui sono scritte
- **Selezione**: permette di scegliere il blocco di istruzioni da eseguire in base al valore di una condizione
- **Iterazione**: permette di eseguire ripetutamente un blocco di istruzioni

Laboratorio di Informatica

Flusso di controllo
Pagina 2

Flusso del controllo

- Le istruzioni di un programma finora sono state eseguite in **sequenza**
- Spesso è però necessario alterare l'ordine sequenziale per:
 - Scegliere tra azioni alternative (**selezione**)
 - `if` , `if - elseif - else` , `switch`
 - Iterare un blocco di istruzioni (**iterazione**)
 - `while` , `for`

Laboratorio di Informatica

Flusso di controllo
Pagina 3

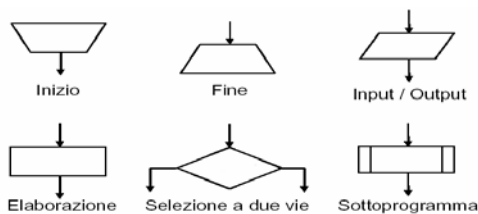
Rappresentazione degli algoritmi

- Per problemi complessi è opportuno pensare prima all'**algoritmo** di risoluzione e scriverlo in modo indipendente dal linguaggio di programmazione.
- Gli algoritmi vengono solitamente rappresentati con due metodologie:
 - **Diagramma di flusso** (grafico)
 - **Pseudo-codice** (simile alla scrittura del programma ma privo di una sintassi rigorosa)

Laboratorio di Informatica

Flusso di controllo
Pagina 4

Diagramma di flusso

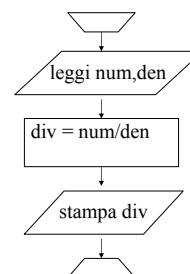


- Diagramma di flusso:
 - Si ottiene collegando diversi elementi grafici con frecce di entrata e di uscita

Laboratorio di Informatica

Flusso di controllo
Pagina 5

Flow-chart e pseudo-codice: divisione tra due numeri

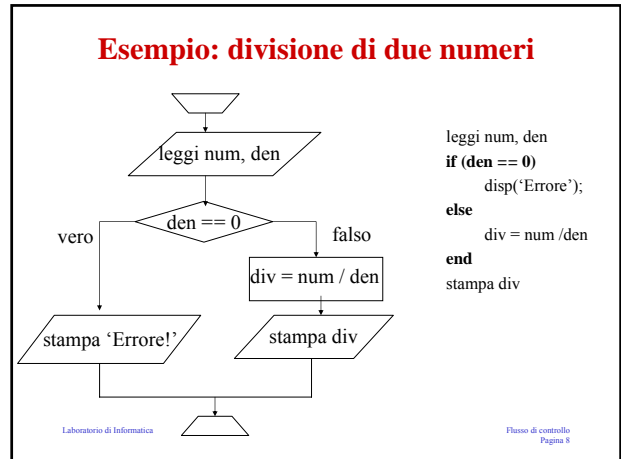
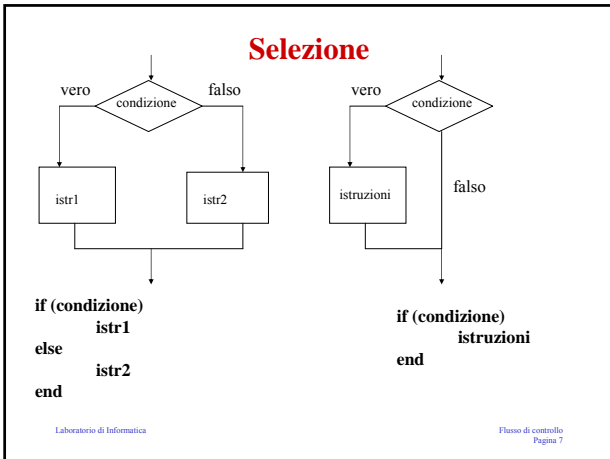


Pseudo-codice

```
leggi num,den  
div = num / den  
stampa div
```

Laboratorio di Informatica

Flusso di controllo
Pagina 6



if annidati

- In una selezione ogni blocco può essere a sua volta una selezione, creando una serie di if annidati

Laboratorio di Informatica Flusso di controllo
Pagina 9

Esercizio: radici equazione

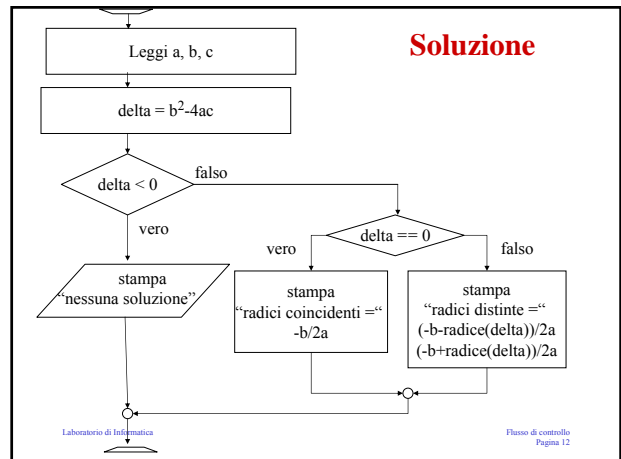
- Argomenti o ingressi o input:**
 - a, b, c** : numeri reali, coefficienti dell'equazione da elaborare
- Risultati o uscite o output:**
 - 'nessuna radice'
 - " $x_1 = x_2 = r$ " se l'equazione ax^2+bx+c ha radici coincidenti = **r**
 - " $x_1 = r_1, x_2 = r_2$ " se l'equazione ax^2+bx+c ha radici distinte = **r1, r2**

Laboratorio di Informatica Flusso di controllo
Pagina 10

L'idea dell'algorithmo di soluzione di RADICI

- Risolvero il problema calcolando il discriminante delta dell'equazione, usando le operazioni di prodotto e differenza;
- analisi poi i vari casi di delta:
 - < 0
 - $= 0$
 - > 0
- e caso per caso costruisco il messaggio da inviare in uscita.
- Il dettaglio si ottiene con un diagramma di flusso:

Laboratorio di Informatica Flusso di controllo
Pagina 11



Pseudo-codice

```
leggi a, b, c
delta = b2 - 4ac
if (delta < 0)
  stampa 'Nessuna radice reale'
else
  if (delta == 0)
    r = -b/2a
    stampa 'Radici coincidenti.' r
  else
    r1 = (-b - sqrt(delta))/2a
    r2 = (-b + sqrt(delta))/2a
    stampa r1, r2
  end
end
```

Espressioni logiche o booleane

- Andiamo ora ad approfondire il concetto intuitivo della “**condizione**”.
- Significa valutare un’*espressione logica*, cioè un’espressione che può assumere solo due valori genericamente chiamati:
 - True (vero)
 - False (falso)
- Esempio: (a < b)

Matlab: Operatori relazionali e d’uguaglianza

- Minore di <
- Maggiore di >
- Minore di o uguale a <=
- Maggiore di o uguale a >=
- Uguale a ==
- Diverso da ~=

Operatori logici

- And &&
- Or ||
- Not ~

```
a = 7;
b = 5;
(a < b) && (b ~= 0) → false
(a < b) || (b ~= 0) → true
~(a < b) → true
```

Esempi

- Scrivere un’espressione logica per:
 - $4 < x < 8$
 - $4 \leq x < 8$
 - $x \neq 0, x > 8$
 - $x \in [0,100], x \notin \{3,5,7,9\}$
 - $x \in (0,100) \vee x \in (200,300)$

Matlab: if-else

- La forma generale è:

```
if (espressione logica)
  istruzioni-1
else
  istruzioni-2
end
```
- Se “espressione” è vera allora vengono eseguite istruzione-1, se “espressione” è falsa vengono eseguite istruzione-2

Matlab: if-else divisione di due numeri

```
num = input('Inserisci numeratore: ');
den = input('Inserisci denominatore: ');

if den == 0
    disp('Errore: denominatore nullo!');
else
    div = num /den;
    disp('Risultato della divisione:');
    disp(div);
end
```

Laboratorio di Informatica

Flusso di controllo
Pagina 19

Matlab: if-else esercizio del minimo

Leggi due numeri a, b e calcolane il minimo

Laboratorio di Informatica

Flusso di controllo
Pagina 20

Matlab: if-else esercizio del minimo

Leggi due numeri a, b e calcolane il minimo

```
a = input('Inserisci un valore: ');
b = input('Inserisci il secondo: ');

disp('Il valore minimo è: ');
if (a < b)
    disp(a);
else
    disp(b);
end
```

Laboratorio di Informatica

Flusso di controllo
Pagina 21

Matlab: elseif

Riprendiamo l'esercizio delle radici e scriviamo in matlab la parte riguardante l'if:

procedura radici

```
leggi a, b, c
delta = b2 -4ac
if (delta < 0)
    stampa "Nessuna radice reale"
else
    if (delta ==0)
        r = -b/2a
        stampa "Radici coinc.:" r
    else
        r1 = (-b - sqrt(delta))/2a
        r2 = (-b + sqrt(delta))/2a
        stampa r1, r2
    end
end
```

Laboratorio di Informatica

procedura radici

```
leggi a, b, c
delta = b2 -4ac
if (delta < 0)
    stampa "Nessuna radice reale"
elseif (delta == 0)
    r = -b/2a
    stampa "Radici coinc.:" r
else
    r1 = (-b - sqrt(delta))/2a
    r2 = (-b + sqrt(delta))/2a
    stampa r1, r2
end
```

NB: l'elseif non richiede un'apposita clausola end

Flusso di controllo
Pagina 22

Matlab: switch

Implementa la **scelta tra diverse alternative** quando la **condizione** discriminante è la **coincidenza** del valore di un'espressione di tipo intero o carattere con un valore dello stesso tipo scelto da un insieme finito

Laboratorio di Informatica

Flusso di controllo
Pagina 23

Matlab: switch

switch espressione

```
case valore-1
    istruzioni-1
case valore-2
    istruzioni-2
...
otherwise
    istruzioni-n
end
```

Laboratorio di Informatica

- Valuta "espressione"
- Salta al case il cui valore coincide con il valore di "espressione"; se non esiste salta a otherwise
- Il otherwise è opzionale
- Al termine prosegue con le istruzioni che seguono end

•NB: anche più valori per case:
case { <val-1>, <val-2>, ...};

Flusso di controllo
Pagina 24

Matlab: switch

Nelle radici abbiamo il problema di non avere costanti ma relazioni.

procedura radici

```
leggi a, b, c
delta = b2-4ac
if (delta < 0)
    stampa "Nessuna radice reale"
elseif (delta ==0)
    r = -b/2a
    stampa "Radici coinc.: " r
else
    r1 = (-b - sqrt(delta))/2a
    r2 = (-b + sqrt(delta))/2a
    stampa r1, r2
end
```

procedura radici

```
absdelta=delta/abs(delta)
switch absdelta
    case -1
        stampa "Nessuna radice reale"
    case 1
        r1 = (-b - sqrt(delta))/2a
        r2 = (-b + sqrt(delta))/2a
        stampa r1,r2
    otherwise
        r = -b/2a
        stampa r
end
```