

Istruzioni condizionali (if-then-else e switch)



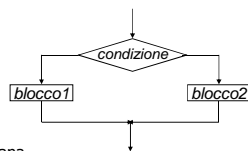
Istruzioni condizionali



Istruzione if

• Sintassi:

```
if (<condizione>)  
  <blocco1>  
[else  
  <blocco2>]
```



<condizione>: Espressione booleana

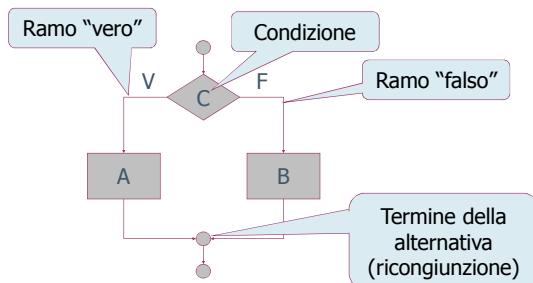
<blocco1,2>: Sequenza di istruzioni

- Se la sequenza contiene più di una istruzione, è necessario racchiuderle tra parentesi graffe

• Significato:

- Se è vera <condizione>, esegui le istruzioni di <blocco1>, altrimenti esegui quelle di <blocco2>

Notazione grafica



Istruzione `if` : Esempio

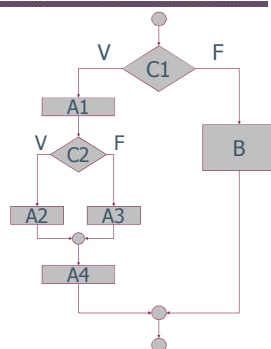
- Leggere due valori A e B, calcolarne la differenza in valore assoluto $D = |A-B|$ e stamparne il risultato

```
main()
{
    int A,B,D;

    scanf("%d %d",&A,&B);
    if (A > B)
        D = A-B;
    else
        D = B-A;
    printf("%d\n",D);
}
```

Scelte annidate

- Nelle istruzioni del blocco "vero" o del blocco "else", è possibile inserire altri blocchi di scelta
- In tal caso la seconda scelta risulta **annidata** all'interno della prima



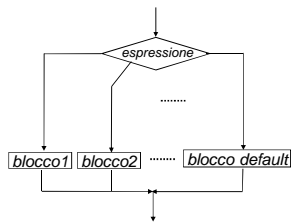
Istruzioni switch



Istruzione switch

- Sintassi:

```
switch (<espressione>)  
{  
    case <costante1>:  
        <blocco1>  
        break;  
    case <costante2>:  
        <blocco2>  
        break;  
    ...  
    default:  
        <blocco default>  
}
```



<espressione>: Espressione a valore numerico

<blocco1>, <blocco2>, ... : Sequenza di istruzioni (no parentesi graffe!)

Istruzione switch (Cont.)

- Significato:

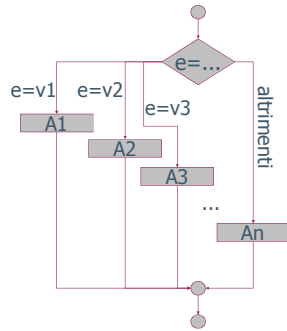
- In base al valore di <espressione>, esegui le istruzioni del case corrispondenti
- Nel caso nessun case venga intercettato, esegui le istruzioni corrispondenti al caso default

- NOTE:

- I vari case devono rappresentare condizioni mutualmente ESCLUSIVE!
- I vari case vengono eseguiti in sequenza
 - Per evitare questo, si usa l'istruzione break all'interno di un blocco

Sintassi istruzione switch

```
switch ( e )
{
    case v1:
        A1 ;
        break ;
    case v2:
        A2 ;
        break ;
    case v3:
        A3 ;
        break ;
    .....
    default:
        An ;
}
```



Istruzione switch: Esempio

```
int x;
...
switch (x) {
    case 1:
        printf("Sono nel caso 1\n");
        break;
    case 2:
        printf("Sono nel caso 2\n");
        break;
    default:
        printf("Né caso 1 né caso 2\n");
        break;
}
```

Rango delle espressioni aritmetiche

- In C, è possibile lavorare con operandi non dello stesso tipo
- Le operazioni aritmetiche avvengono dopo aver promosso tutti gli operandi al tipo di rango più alto:

↓

```
_Bool
char
short
unsigned short
int
unsigned int
long
unsigned long
long long
unsigned long long
float
double
long double
```

Operatori di cast

- In alcuni casi, può essere necessario convertire esplicitamente un'espressione in uno specifico tipo
 - Quando le regole di conversione automatica non si applicano
 - Esempio: `int i; double d;`
l'assegnazione `i = d;` fa perdere informazione
- Sintassi:
'(' <tipo> ')' <espressione>;
 - Significato: Forza <espressione> ad essere interpretata come se fosse di tipo <tipo>
- Esempio:
...
`double f;`
`f = (double) 10;`

13

Operatori di cast: Esempio

```
#include <stdio.h>

main()
{
    int a, b;

    printf("Dammi un numero intero (A): ");
    scanf("%d",&a);
    printf("Dammi un numero intero (B): ");
    scanf("%d",&b);
    if(b==0)
        printf("Errore: divisione per zero!!\n");
    else
        printf("A / B = %f\n", ((float)a)/b);
}
```

14

Operatore sizeof()

- E' possibile calcolare il numero di byte utilizzato dai tipi di dato di base utilizzando l'operatore `sizeof`
- Sintassi:
`sizeof(<tipo>)`
- Ritorna il numero di byte occupato da <tipo>
- Esempio:
`unsigned size;`
`size = sizeof(float); /* size = 4 */`
- L'uso dell'operatore `sizeof()` può essere esteso al calcolo dello spazio occupato da espressioni, vettori e strutture

15

Operatore sizeof(): Esempio

```
#include <stdio.h>

main()
{
    printf("tipo      n.byte\n");
    printf("-----\n");
    printf("char      %d\n", sizeof(char));
    printf("int       %d\n", sizeof(int));
    printf("long      %d\n", sizeof(long));
    printf("long long %d\n", sizeof(long long));
    printf("float     %d\n", sizeof(float));
    printf("double    %d\n", sizeof(double));
    printf("long double %d\n", sizeof(long double));
}
```

16

Fine Capitolo

