

Laboratorio di informatica

Ingegneria meccanica

Esercitazione 3 - 17 ottobre 2007

1

Operatori logici in C (I)

- Per le operazioni logiche di **AND**, **OR** e **NOT** sono disponibili, rispettivamente, gli operatori **&&**, **||** e **!**
- Gli operandi di un operatore logico vengono resi logici in questo modo: il valore **0** viene interpretato come **FALSO**, mentre un valore diverso da **0** (qui indicato con **≠0**) viene interpretato come **VERO**
- **&&**, **||** e **!** forniscono come risultato **0** (inteso come **FALSO**) o **1** (inteso come **VERO**)

A	B	A && B	A B	!A
0	0	0	0	1
0	≠0	0	1	
≠0	0	0	1	0
≠0	≠0	1	1	

2

Operatori logici in C (II)

A	B	A && B	A B	!A
0	0	0	0	1
0	≠0	0	1	
≠0	0	0	1	0
≠0	≠0	1	1	

- La valutazione delle espressioni contenenti **&&** e **||** obbedisce alle regole seguenti
- **espressione1 && espressione2**: Se **espressione1** vale **0 (FALSO)**, **espressione2** non viene valutata e l'espressione complessiva vale **0 (FALSO)**
- **espressione1 || espressione2**: Se **espressione1** vale **≠0 (VERO)**, **espressione2** non viene valutata e l'espressione complessiva vale **1 (VERO)**

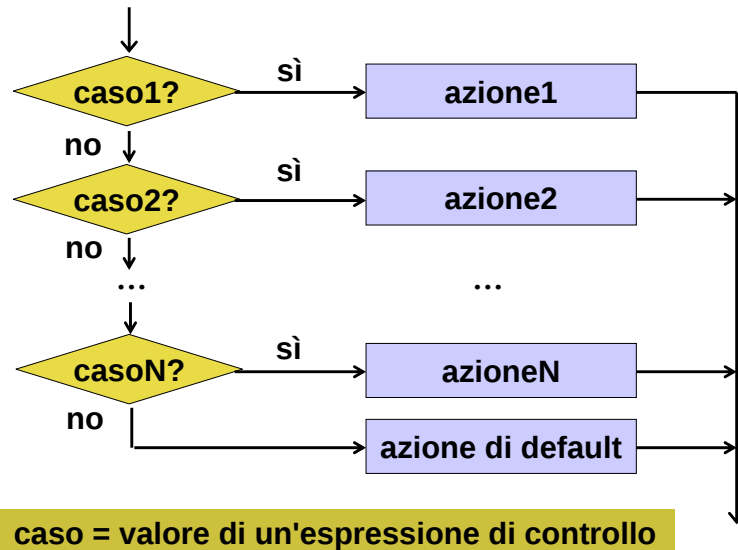
3

Strutture di selezione in C

- **Selezione singola if**: viene eseguita o meno una singola azione
- **Selezione doppia if else**: viene eseguita una tra due azioni *logicamente* distinte (caso particolare: una delle due azioni è nulla → selezione singola)
- **Selezione multipla switch**: viene eseguita una tra un certo numero di azioni *logicamente* distinte
- Un'azione è formulata con un blocco di istruzioni
- La selezione è sempre in base al valore di un'espressione, distinguendo solo due casi: **zero** e **diverso da zero**

4

Selezione multipla



5

C: Istruzione **switch** (1)

```
switch ( espressione )
{
    case costante1 : sequenza-istruzioni1 ; break ;
    case costante2 : sequenza-istruzioni2 ; break ;
    ...
    case costanteN : sequenza-istruzioniN ; break ;
    default : sequenza-istruzioni ; break ;
}
```

Se il valore di **espressione** *uguaglia* una delle **costanti** specificate, allora si esegue la **sequenza-istruzioni** corrispondente, altrimenti si esegue la sequenza-istruzioni di **default**

6

C: Istruzione **switch** (2)

- In **switch**, una **costante** è un intero valido o un carattere (si usa in questo l'intero associato)
- Le **costanti** specificate devono avere valori *distinti*
- In **switch**, una **sequenza-istruzioni** non richiede l'uso di parentesi graffe
- **break ;** provoca l'*uscita* dalla struttura

7

Esempio (1/2)

```
#include <stdio.h>
```

```
int main( ) {
```

```
    int opzione;
```

```
    float tc, tf, conv;
```

```
    printf ("Premi \n'1' per trasformare da F a C\n'2' per  
trasformare da C a F\nScelta: ");
```

```
    scanf("%d", &opzione);
```

```
    switch (opzione) {
```

```
        case 1:
```

```
            conv = 5./9.;
```

```
            printf("\nValore in gradi Fahrenheit = ");
```

```
            scanf("%f", &tf);
```

```
            tc = (tf - 32) * conv;
```

```
            printf("Valore in gradi Celsius = %f\n", tc);
```

```
            break;
```

8

Esempio (2/2)

case 2:

```
conv = 9./5.;
printf("\nValore in gradi Celsius = ");
scanf("%f", &tc);
tf = tc * conv + 32;
printf("Valore in gradi Fahrenheit = %f\n", tf);
break;
```

default:

```
printf ("Opzione non prevista\n");
break;
```

```
}
return 0;
}
```

9

Esercizio 1

Scrivere un programma che legga da standard input due numeri interi e visualizzi su standard output una tabella con i risultati delle operazioni logiche AND, OR, NOT

```
Inserire il primo numero:
7
Inserire il secondo numero:
0

A B A&B A||B !A
7 0 0 1 0

Premere un tasto per continuare . . .
```

10

Esercizio 2: data valida

Scrivere un programma che legga da standard input tre numeri interi non negativi da intendersi come giorno, mese, anno, e stampi un messaggio in cui si dice se i tre numeri rappresentano o no una data valida (si escluda il caso di anno bisestile)

USARE **switch** e **operatori logici**

11

Esercizio 3: data giorno successivo

Scrivere un programma che legge da standard input tre numeri interi che rappresentano una data (giorno, mese e anno) e stampa tre interi che rappresentano la data del giorno successivo

- Tenere conto del fatto che i mesi possono essere di 28 (o 29 negli anni **bisestili**), 30 e 31 giorni
- Anno **bisestile**: **divisibile per 400** oppure **divisibile per 4 e non divisibile per 100**. Ad esempio, il 1996 è un anno bisestile, dato che è divisibile per 4 e non è divisibile per 100. L'anno 2000 è un anno bisestile, dato che è divisibile per 400

12

Algoritmo: data giorno successivo (1)

1. Leggi giorno, mese, anno
2. Determina numero giorni mese
Se mese è 1,3,5,7,8,10,12: giornimese = 31;
Se mese è 2: se anno è bisestile,
 giornimese = 29, altrimenti giornimese = 28;
Se mese è 4, 6, 9, 11: giornimese = 30;
Diversamente: mese errato

13

Algoritmo: data giorno successivo (2)

3. Determina data giorno successivo
Se mese errato oppure giorno errato:
 stampa messaggio d'errore
Altrimenti, se giorno diverso da giornimese:
 data giorno successivo = (giorno+1, mese, anno)
Altrimenti, se mese diverso da 12:
 data giorno successivo = (1, mese+1, anno)
Altrimenti:
 data giorno successivo = (1, 1, anno+1)
4. Stampa data giorno successivo

14