



PEGASO
Università Telematica



Indice

1. ISTRUZIONE SWITCH	3
2. DETTAGLI DELL'ISTRUZIONE SWITCH.....	7
3. DIAGRAMMA DI FLUSSO DELL'ISTRUZIONE SWITCH	9
RIFERIMENTI BIBLIOGRAFICI	12

1. Istruzione switch

Abbiamo esaminato l'istruzione di selezione singola `if` e l'istruzione di selezione doppia `if...else`.

A volte, un algoritmo conterrà una serie di decisioni in cui una variabile o un'espressione viene testata separatamente per ognuno dei valori interi costanti che può assumere e per i quali vengono intraprese azioni differenti.

Tale situazione è chiamata selezione multipla.

Per trattare questo processo decisionale, il C fornisce l'istruzione di selezione multipla `switch`, che consiste in una serie di etichette `case`, un cosiddetto caso default opzionale e istruzioni da eseguire per ognuno dei `case`.

Il seguente programma usa l'istruzione `switch` per contare il numero di volte che ogni singolo voto a lettera 2 è stato ottenuto dagli studenti in un esame.

```
01 // Programma C
02 // Conteggio dei voti a lettera con switch.
03 #include <stdio.h>
04
05 int main(void)
06 {
07     unsigned int aCount = 0;
08     unsigned int bCount = 0;
09     unsigned int cCount = 0;
10     unsigned int dCount = 0;
11     unsigned int fCount = 0;
12
13     puts("Enter the letter grades.");
14     puts("Enter the EOF character to end input.");
15     int grade; // un voto
16
17     // ripeti finché l'utente non immette la sequenza di end-of-file
18     while ((grade = getchar()) != EOF) {
19
```

Filippo Cugini - Istruzione di selezione multipla switch

```
20 // determina quale voto e' stato inserito
21 switch (grade) { // switch annidato nel while
22
23     case 'A': // il voto e' la lettera maiuscola A
24     case 'a': // o la lettera minuscola a
25         ++aCount;
26         break; // necessario per uscire dallo switch
27
28     case 'B': // il voto e' la lettera maiuscola B
29     case 'b': // o la lettera minuscola b
30         ++bCount;
31         break;
32
33     case 'C': // il voto e' la lettera maiuscola C
34     case 'c': // o la lettera minuscola c
35         ++cCount;
36         break;
37
38     case 'D': // il voto e' la lettera maiuscola D
39     case 'd': // o la lettera minuscola d
40         ++dCount;
41         break;
42
43     case 'F': // il voto e' la lettera maiuscola F
44     case 'f': // o la lettera minuscola f
45         ++fCount;
46         break;
47
48     case '\n': // ignora i newline,
49     case '\t': // le tabulazioni
50     case ' ': // e gli spazi in input
51         break;
52
53     default: // cattura tutti gli altri caratteri
54         printf("%s", "Incorrect letter grade entered.");
55         puts(" Enter a new grade.");
56         break; // opzionale; uscirà comunque dallo switch
57 }
58 } // fine di while
```

```
59
60 // stampa il riepilogo dei risultati
61 puts("\nTotals for each letter grade are:");
62 printf("A: %u\n", aCount);
63 printf("B: %u\n", bCount);
64 printf("C: %u\n", cCount);
65 printf("D: %u\n", dCount);
66 printf("F: %u\n", fCount);
67 }
```

Enter the letter grades.

Enter the EOF character to end input.

a

b

c

C

A

d

f

C

E

Incorrect letter grade entered. Enter a new grade.

D

A

b

^Z _____Non tutti i sistemi mostrano una rappresentazione del carattere EOF

Totals for each letter grade are:

A: 3

B: 2

C: 3

D: 2

F: 1

2. Dettagli dell'istruzione switch

La parola chiave `switch` è seguita dal nome della variabile `grade` tra parentesi.

Questa è chiamata espressione di controllo.

Il valore di questa espressione è confrontato con ognuna delle etichette `case`.

Supponete che l'utente abbia inserito la lettera C come valore per `grade`.

Il carattere C è automaticamente confrontato con ogni `case` nello `switch`.

Se il confronto ha successo (`case 'C':`), vengono eseguite le istruzioni per quel `case`.

Nel caso della lettera C, `cCount` è incrementato di 1 (riga 35) e si esce immediatamente dall'istruzione `switch` con l'istruzione `break`.

L'istruzione `break` fa sì che il controllo del programma continui con la prima istruzione dopo l'istruzione `switch`.

Viene usata l'istruzione `break` perché i vari `case` in un'istruzione `switch` verrebbero altrimenti eseguiti tutti insieme.

Se non viene usato `break` da nessuna parte in un'istruzione `switch`, ogni volta che si verificherà un confronto positivo nell'istruzione verranno eseguite le istruzioni per tutti i `case` rimanenti.

Se non si verificano confronti positivi, viene eseguito il caso default e, in questo programma, viene stampato il messaggio di "Incorrect Letter".

☹ *Errore comune di programmazione*

Dimenticare un'istruzione `break` quando è necessaria in un'istruzione `switch` è un errore logico.

😊 *Prevenzione di errori*

Inserite un caso default nelle istruzioni `switch`.

Filippo Cugini - Istruzione di selezione multipla switch

I valori non esplicitamente testati in uno *switch* sarebbero normalmente ignorati.

Il caso *default* vi aiuta a evitare che ciò accada facendo focalizzare la vostra attenzione sulla necessità di processare condizioni eccezionali.

Talvolta le elaborazioni di *default* non sono necessarie.

☺ Buona pratica di programmazione

Benché le clausole *case* e la clausola del caso *default* in un'istruzione *switch* possano presentarsi in un ordine qualsiasi, è uso comune porre la clausola di *default* per ultima.

☺ Buona pratica di programmazione

In un'istruzione *switch* nella quale la clausola di *default* è ultima, l'istruzione *break* finale non è necessaria.

Potreste voler includere questo *break* per chiarezza e simmetria con gli altri *case*.

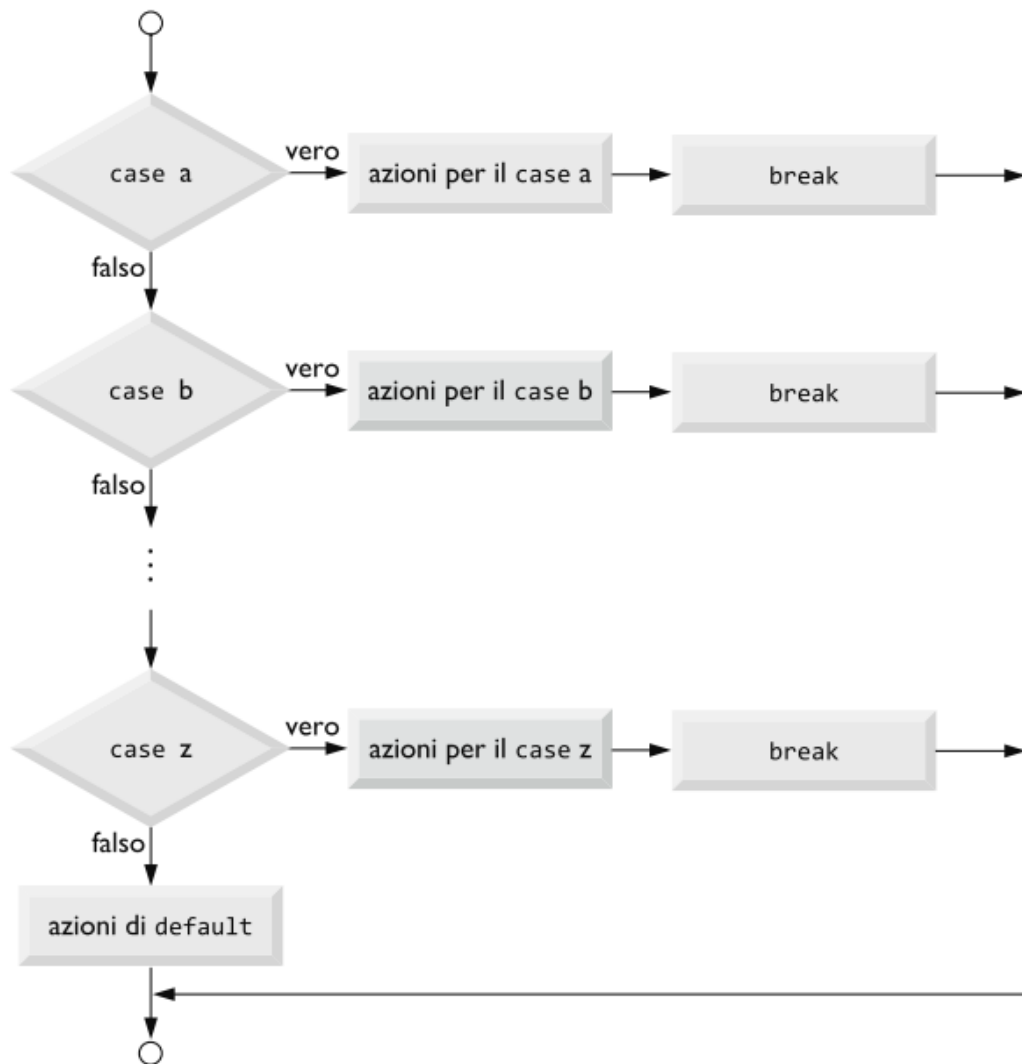
3. Diagramma di flusso dell'istruzione switch

Ogni case può avere una o più azioni.

L'istruzione `switch` è differente da tutte le altre istruzioni di controllo per il fatto che non sono necessarie le parentesi graffe attorno alle sequenze di azioni in un `case`.

L'istruzione di selezione multipla generale `switch` (che usa un `break` in ogni `case`) è rappresentata dal diagramma di flusso seguente.

Il diagramma di flusso rende chiaro che ogni istruzione `break` alla fine di un `case` causa l'uscita immediata dall'istruzione `switch`.



Ignorare i caratteri di newline, tabulazione e spaziatura nell'input

Nell'istruzione `switch`, le righe

```
case '\n': // ignora i newline,  
case '\t': // le tabulazioni  
case ' ': // e gli spazi in input  
    break;
```

fanno sì che il programma salti i caratteri di newline, tabulazione e spazio.

La lettura dei caratteri uno alla volta può causare alcuni problemi.

Per far sì che un programma legga i caratteri dovete inviarli al computer premendo Invio.

Questo fa sì che venga inserito nell'input il carattere newline dopo il carattere che desideriamo processare.

Spesso, questo carattere newline deve essere opportunamente ignorato per far funzionare il programma in modo corretto.

I case precedenti nella nostra istruzione `switch` impediscono che il messaggio di errore nel caso default sia stampato ogni volta che nell'input si incontra un newline, una tabulazione o uno spazio.

Pertanto, ciascun input causa due iterazioni del ciclo: la prima per il voto a lettera e la seconda per '\n'.

Scrivere diverse etichette case insieme (come `case 'D': case 'd':`) significa semplicemente che lo stesso insieme di azioni viene eseguito per l'uno o l'altro di questi case.

Quando usate l'istruzione `switch`, ricordate che ogni singolo case può testare solamente un'espressione costante intera, cioè qualsiasi combinazione di costanti di tipo carattere e di costanti intere il cui valore è un valore costante intero.

Una costante di tipo carattere può essere rappresentata come un carattere specifico tra virgolette singole, come ad esempio 'A'.

I caratteri devono essere racchiusi entro virgolette singole per essere riconosciuti come costanti di tipo carattere (i caratteri tra virgolette doppie sono riconosciuti come stringhe). Le costanti intere sono semplicemente valori interi.

Riferimenti bibliografici

- Paul Deitel, Harvey Deitel, "Il linguaggio C – Fondamenti e tecniche di programmazione",
Libro edito da Pearson Italia. Include anche utili esercizi di autovalutazione.