



REPORT DEBUG

Nella prima parte vedremo il codice iniziale con dei commenti accanto per capire esattamente cosa fa il programma. Successivamente, vedremo delle slide evidenziando tutti gli errori e come si potrebbero correggere. Alla fine, presenterò una versione del codice corretta e ottimizzata.



```
1  #include <stdio.h> //libreria
2
3  void menu();        // Crea una funzione vuota menu
4  void moltiplica();  // Crea una funzione vuota moltiplica
5  void dividi();      // Crea una funzione vuota dividi
6  void ins_string();  // Crea una funzione vuota ins_string
7
8  int main() // Funzione principale MAIN
9
10 {
11     char scelta = {'\0'}; // Variabile per dare una scelta all'utente
12     menu();                // Richiama la funzione menu
13     scanf("%d", &scelta); // Chiedo all'utente un input
14
15     switch (scelta) // Creo e implemento uno Switch con i relativi casi in base alla scelta dell'utente
16     {
17     case 'A': // Caso A = Chiama la funzione moltiplica
18         moltiplica();
19         break;
20     case 'B': // Caso B = Chiama la funzione dividi
21         dividi();
22         break;
23     case 'C': // Caso C = Chiama la funzione ins_string
24         ins_string();
25         break;
26     }
27
28     return 0;
29 }
```

```

30
31 void menu() // Crea la funzione menu e implementa una logica
32 {
33     printf("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n"); // stampa a schermo delle info
34     printf("Come posso aiutarti?\n"); // stampa a schermo delle info
35     printf("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n"); // stampa a schermo delle info
36 }
37
38 void moltiplica() // Crea la funzione moltiplica e implementa una logica
39 {
40     short int a, b = 0; // crea due variabili "short e int" che per lui saranno uguali a 0
41     printf("Inserisci i due numeri da moltiplicare:"); // stampa a schermo delle info
42     scanf("%f", &a); // Chiede all'utente di inserire un input di tipo FLOAT
43     scanf("%d", &b); // Chiede all'utente di inserire un input di tipo INT
44
45     short int prodotto = a * b; // crea la logica per far si che la variabile prodotto faccia la moltiplicazione tra due variabili
46
47     printf("Il prodotto tra %d e %d e': %d", a, b, prodotto); // stampa le info
48 }
49

```

```

50 void dividi() // Crea la funzione dividi e implementa una logica
51 {
52     int a, b = 0; // crea due variabili int pensando di inizializarle a 0
53     printf("Inserisci il numeratore:"); // stampa le info
54     scanf("%d", &a); // input utente di tipo INT
55     printf("Inserisci il denominator:"); // stampa le info
56     scanf("%d", &b); // input utente di tipo INT
57
58     int divisione = a % b; // crea la logica per far si che la varibile Divisione faccia effettivamente la ""divisione"" tra due variabili
59
60     printf("La divisione tra %d e %d e': %d", a, b, divisione); // stampa le info
61 }
62
63 void ins_string() // Crea la funzione ins_string e implementa una logica
64 {
65     char stringa[10]; // crea una variabile String che possa contenere massimo 9 caratteri + 1 carattere terminatore
66     printf("Inserisci la stringa:"); // stampa
67     scanf("%s", &stringa); // input utente
68 }
69

```

Evidenzio gli errori del codice



Sono state dichiarate funzioni void senza istruzioni:
non ha senso.

Cancellerei queste righe di codice in quanto le
creeremo più avanti.

La variabile di tipo CHAR non può
essere inizializzata in questo modo.

Cancellerei dal segno '=' fino alla fine
della parentesi graffa.

Manca una parte del codice allo switch per
garantire il corretto funzionamento della logica.

Aggiungerei il caso di DEFAULT per gestire
qualsiasi altra eventualità che l'utente potrebbe
far verificare.

```
1  #include <stdio.h>
2
3  void menu();
4  void multiplica();
5  void dividi();
6  void ins_string();
7
8  int main()
9
10 {
11     char scelta = {'\0'};
12     menu();
13     scanf("%d", &scelta);
14
15     switch (scelta)
16     {
17     case 'A':
18         multiplica();
19         break;
20     case 'B':
21         dividi();
22         break;
23     case 'C':
24         ins_string();
25         break;
26     }
27
28     return 0;
29 }
```

In C, non possiamo inizializzare più variabili contemporaneamente in questa maniera.

Nel **scanf** stiamo indicando che il primo numero sarà un **float**.

Non possiamo inizializzare la variabile **prodotto** in due modi diversi.

Nel **printf** stiamo specificando che ogni variabile che stamperemo sarà di tipo **int**

Inizializzerei le due variabili con il tipo **int** senza fornire loro un valore di partenza.

Nel **scanf**, indicherei che entrambi i numeri saranno di tipo **int**, utilizzando **%d**.

Cambierei la dichiarazione della variabile **PRODOTTO**, rendendola di tipo **int**.

Il **printf**, mantenendo invariati i valori precedenti, rimarrebbe inalterato.

```
30
31 void menu()
32 {
33     printf("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
34     printf("Come posso aiutarti?\n");
35     printf("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
36 }
37
38 void moltiplica()
39 {
40     short int a, b = 0;
41     printf("Inserisci i due numeri da moltiplicare:");
42     scanf("%f", &a);
43     scanf("%d", &b);
44
45     short int prodotto = a * b;
46
47     printf("Il prodotto tra %d e %d e': %d", a, b, prodotto);
48 }
49
50 void dividi()
51 {
52     int a, b = 0;
53     printf("Inserisci il numeratore:");
54     scanf("%d", &a);
55     printf("Inserisci il denominatore:");
56     scanf("%d", &b);
57
58     int divisione = a % b;
59
60     printf("La divisione tra %d e %d e': %d", a, b, divisione);
61 }
62
63 void ins_string()
64 {
65     char stringa[10];
66     printf("Inserisci la stringa:");
67     scanf("%s", &stringa);
68 }
69
```


Le variabili **a** e **b** non possono essere inizializzate in questa maniera.

La variabile **divisione** non restituirà il risultato della divisione, ma darà la differenza del risultato.

Poiché la divisione è un'operazione su numeri interi, potrebbe produrre un risultato poco preciso.

Io inizierei le variabili in **FLOAT** per avere un risultato più accurato, e poi non assegnerei loro un valore iniziale.

Cambierei il simbolo **%** con **/** sulla variabile divisione per ottenere il risultato desiderato.

Modificherei i **%d (interi)** con **%f (float)** per ottenere un risultato più accurato.

```
30
31 void menu()
32 {
33     printf("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
34     printf("Come posso aiutarti?\n");
35     printf("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
36 }
37
38 void moltiplica()
39 {
40     short int a, b = 0;
41     printf("Inserisci i due numeri da moltiplicare:");
42     scanf("%f", &a);
43     scanf("%d", &b);
44
45     short int prodotto = a * b;
46
47     printf("Il prodotto tra %d e %d e': %d", a, b, prodotto);
48 }
49
50 void dividi()
51 {
52     int a, b = 0;
53     printf("Inserisci il numeratore:");
54     scanf("%d", &a);
55     printf("Inserisci il denominatore:");
56     scanf("%d", &b);
57
58     int divisione = a % b;
59
60     printf("La divisione tra %d e %d e': %d", a, b, divisione);
61 }
62
63 void ins_string()
64 {
65     char stringa[10];
66     printf("Inserisci la stringa:");
67     scanf("%s", &stringa);
68 }
69
```

In questo blocco di istruzioni non c'è nessun vero errore, ma solo una mancanza di sicurezza nella stampa dei dati e la stampa dei dati stessa.

Io incrementerei il limite massimo di caratteri della stringa di 1, per dare spazio a effettivamente 10 caratteri l'undicesimo sarà il carattere terminatore nullo '1n'

Nello scanf modificherei %s aggiungendo la specifica %10s per limitare la lettura a un massimo di 10 caratteri, al fine di evitare una potenziale condizione di overflow del buffer.

Alla fine, aggiungerei un printf per stampare a schermo l'input dell'utente.

```
30
31 void menu()
32 {
33     printf("Benvenuto, sono un assistente digitale, posso aiutarti
34     printf("Come posso aiutarti?\n");
35     printf("A >> Moltiplicare due numeri\nB >> Dividere due numeri
36 }
37
38 void moltiplica()
39 {
40     short int a, b = 0;
41     printf("Inserisci i due numeri da moltiplicare:");
42     scanf("%f", &a);
43     scanf("%d", &b);
44
45     short int prodotto = a * b;
46
47     printf("Il prodotto tra %d e %d e': %d", a, b, prodotto);
48 }
49
50 void dividi()
51 {
52     int a, b = 0;
53     printf("Inserisci il numeratore:");
54     scanf("%d", &a);
55     printf("Inserisci il denominatore:");
56     scanf("%d", &b);
57
58     int divisione = a % b;
59
60     printf("La divisione tra %d e %d e': %d", a, b, divisione);
61 }
62
63 void ins_string()
64 {
65     char stringa[10];
66     printf("Inserisci la stringa:");
67     scanf("%s", &stringa);
68 }
69
```


Codice rivisitato



```

1  #include <stdio.h>
2
3  void menu() // Funzione menu
4  {
5      printf("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n"); // Stampo i dati*3
6      printf("Come posso aiutarti?\n");
7      printf("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
8  }
9
10 void moltiplica() // Funzione moltiplicazione
11 {
12     int a, b; // variabili di interi
13     printf("Inserisci i due numeri da moltiplicare:");
14     scanf("%d %d", &a, &b); // input utente
15
16     int prodotto = a * b; // la variabile prodotto fara la moltiplicazione tra due variabili/numeri
17
18     printf("Il prodotto tra %d e %d e': %d\n", a, b, prodotto); // stampo a schermo il risultato
19 }
20

```

```

20
21 void dividi() // Funzione divisione
22 {
23     float a, b; // creo due variabili di tipo FLOAT
24     printf("Inserisci il numeratore:");
25     scanf("%f", &a); // input di tipo float*2
26     printf("Inserisci il denominatore:");
27     scanf("%f", &b);
28
29     if (b != 0) // creo il controllo IF che un numero non può essere diviso per 0
30     {
31         float divisione = a / b; // la variabile divisione farà la divisione tra due variabili/numeri
32
33         printf("La divisione tra %.2f e %.2f e': %.2f\n", a, b, divisione); // stampo a schermo il risultato
34     }
35     else
36     {
37         printf("Impossibile dividere per zero.\n"); // stampo a schermo l'errore se si prova a dividere per 0
38     }
39 }

```

```
41
42 void ins_string() // funzione ins_string
43 {
44
45     char stringa[11];      //variabile stringa con lunghezza massima di caratteri = 10 + 1 carattere terminale
46     printf("Inserisci la stringa:");      //stampo info
47     scanf("%10s", stringa);      //input con massimo di "salvataggio" di 10 caratteri grazie al %10s
48
49     printf("Hai inserito la stringa: %s\n", stringa);      //stampo le info
50 }
51
```

```
52
53 int main()      //Funzione principale
54 {
55
56     char scelta;      //variabile scelta
57
58     menu();      //richiamo la funzione menu
59     scanf(" %c", &scelta);      //input utente per la scelta
60
61     switch (scelta)      //implemento la gestione delle scelte con lo Switch
62     {
63     case 'A':      //caso A richiama la funzione moltiplica
64         moltiplica();
65         break;
66     case 'B':      //caso B richiama la funzione dividi
67         dividi();
68         break;
69     case 'C':      //caso C richiama la funzione ins_string
70         ins_string();
71         break;
72     default:      //se nessun caso e vero partira il Default
73         printf("Scelta non valida.\n");
74         return 1;
75     }
76
77     return 0;
78 }
```