

```

1 #include <stdio.h>

2 void menu ();
3 void moltiplica ();
4 void dividi ();
5 void ins_string();
6
7
8 int main ()

9{
    10 char scelta = {'\0'},
    11 menu ();
    12 scanf ("%d", &scelta);
13
    14 switch (scelta)
    15 {
    16     case 'A':
    17         moltiplica();
    18         break;
    19     case 'B':
    20         dividi();
    21         break;
    22     case 'C':
    23         ins_string();
    24         break;
25     }

26 return 0;
27
28 }
29
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
39
40 void moltiplica ()
41 {
42     short int a,b = 0;
43     printf ("Inserisci i due numeri da moltiplicare:");
44     scanf ("%f", &a);
45     scanf ("%d", &b);
46
47     short int prodotto = a * b;
48
49     printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto);

```

```

50 }
51
52
53 void dividi ()
54 {
55     int a,b = 0;
56     printf ("Inserisci il numeratore:");
57     scanf ("%d", &a);
58     printf ("Inserisci il denominatore:");
59     scanf ("%d", &b);
60
61     int divisione = a % b;
62
63     printf ("La divisione tra %d e %d e': %d", a,b,divisione);
64 }
65
66
67
68
69
70 void ins_string ()
71 {
72     char stringa[10];
73     printf ("Inserisci la stringa:");
74     scanf ("%s", &stringa);
75 }

```

```

-----
-----
-----

```

RELAZIONE E ANALISI.

- Capire cosa fa il programma senza eseguirlo

Questo esercizio ci permette di moltiplicare due numeri, gli calcola e stampa il risultato, ci permette anche la divisione di due numeri calcolando il risultato e stampandolo. Gestisce anche Se il denominatore è 0 con la visualizzazione di un messaggio. Il programma ci permette anche di inserire una stringa di massimo 10 caratteri (0-9).

- Individuare dal codice sorgente le casistiche non standard che il programma non gestisce (esempio, comportamenti potenziali che non sono stati contemplati).

10 char scelta = {'\0'}; //non è corretto far risultare un char come una stringa, all'utente serve digitare solo un carattere. La correzione giusta è: char scelta = ('');

12 scanf ("%d", &scelta); // In questo caso %d(decimale) è sbagliato perchè bisogna inserire %c(carattere) perchè all'utente serve digitare un carattere e non un numero.

24 break; // Sotto Break va inserito Default perchè se l'utente schiaccia un carattere non memorizzato, il programma si chiuderà ed è un errore - Errore Logico-.

short int a,b = 0; // In questo caso il programma ci restituirà un risultato di interi senza la virgola, per far scrivere al programma che vogliamo visualizzare i numeri calcolati dopo la virgola bisogna inserire Float, quindi scriviamo " float int a,b = 0;"

44 scanf ("%d", &b); // In questo caso %d in parentesi ci indica un valore decimale quindi non con la virgola. Bisognerebbe modificare %d con %f(funzione).

49 printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto); // anche qui come detto in precedenza, bisognerebbe sostituire %d con %f, in modo tale che l'utente moltiplica numeri con la virgola. Questo è un errore logico.

55 int a,b = 0; // in questo caso il denominatore (b) è 0 e quindi in una divisione quando si ha il denominatore 0 è impossibile. Bisogna inserire la condizione b!=0.

57 scanf ("%d", &a); // In questo caso stesso ragionamento della moltiplicazione, %d indica che deve essere letto come valore intero non calcolando i numeri con la virgola. Bisognerebbe andare a modificarlo con %f per stamparlo con numeri con la virgola. -errore di sintassi-

61 int divisione = a % b; // In questo caso è stata sbagliata la metodologia della divisione che si indica con il simbolo /. In questo caso andremo a digitare int divisione = a/b; -errore logico-

74 scanf ("%s", &stringa); // In questa ultima parte di codice perchè bisognerebbe evitare lo stack overflow che succede quando si utilizza una quantità eccessiva di memoria dello stack. Si dovrebbe modificare lo scanf inserendo delle virgolette "%s", stringa. Questo ci permette di non dover utilizzare più di dieci caratteri e non rischiamo il crash del programma.