

Debuggare Esercizio_5_Epicode_w2

Tipo di programma

Individuazione dal codice sorgente le casistiche non standard che il programma non gestisce

Individuazione di errori di sintassi/logici

```

GNU nano 6.3      Esercizio_5_Epicode_w2.c *
1 #include <stdio.h> //dalla lettura del codice, il programma
2 //possiede diversi errori di sintassi e logici
3 void menu ();      //che non lo portano a svolgere
4 void moltiplica (); //le due operazioni: moltiplicazione, divisione
5 void dividi ();    //e a non generare una stringa di dieci caratteri
6 void ins_string();
7
8
9 int main ()
10 {
11     char scelta = {'\0'};
12     menu ();
13     scanf ("%d", &scelta);
14
15     switch (scelta)
16     {
17         case 'A':
18             moltiplica();
19             break;
20         case 'B':
21             dividi();
22             break;
23         case 'C':
24             ins_string();
25             break;
26     }
27     return 0;
28 }
29
30

```

```

GNU nano 6.3      Esercizio_5_Epicode_w2.c *
1 #include <stdio.h> //direttiva del processore per caricare in memoria la libreria <std.io>
2
3 void menu ();      //funzioni void che non restituiscono
4 void moltiplica (); //nessun risultato
5 void dividi ();    //infatti sono prive
6 void ins_string(); //del valore di ritorno
7
8
9 int main ()        /*funzione principale di ogni programma dalla quale
10                    //il processore parte a leggere le istruzioni all'esecuzione*/
11 {
12     char scelta = '\0'; //variabile per l'inserimento dei caratteri
13     menu ();
14     scanf ("%c", &scelta); /*funzione per leggere l'input da tastiera
15                            //input di tipo char assegnato alla variabile scelta */
16     switch (scelta)    //istruzione condizionale per la scelta multipla del menu
17     {
18         case 'A':      //prima scelta del programma
19             moltiplica();
20             break;     //istruzione per uscire dal set eseguito
21         case 'B':
22             dividi();
23             break;
24         case 'C':
25             ins_string();
26             break;
27         default:       //blocco di istruzione in caso di mancata assunzione dei valori di (scelta)
28             printf ("\nInserire A, B o C\n");
29             break;
30     }
31     return 0;         //valore di ritorno di int main()
32 }
33

```

Debugging Esercizio_5_Epicode_w2 (part. 1)

Analisi da Riga 1 a Riga 30/33

L'immagine a destra mostra il programma originale, l'immagine a sinistra il programma debuggato.

```

GNU nano 6.3      Esercizio_5_Epicode_w2.c *
30
31
32 void menu ()
33 {
34     printf ("Benvenuto, sono un assistente digitale;\n");
35     printf ("posso aiutarti a sbrigare alcuni compiti\n");
36     printf ("Come posso aiutarti?\n");
37     printf ("\nA >> moltiplicare due numeri\n");
38     printf ("B >> Dividere due numeri\n");
39     printf ("C >> Inserire una stringa\n");
40 }
41
42 void moltiplica ()
43 {
44     short int a,b = 0;
45     printf ("Inserisci i due numeri da moltiplicare:");
46     scanf ("%f", &a);
47     scanf ("%d", &b);
48
49     short int prodotto = a * b;
50
51     printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto);
52 }
53

```

```

GNU nano 6.3      Esercizio_5_Epicode_w2.c *
33
34
35 void menu ()          //funzione per la stampa a video del menu principale
36 {
37     printf ("Benvenuto, sono un assistente digitale;\n"); //funzioni
38     printf ("posso aiutarti a sbrigare alcuni compiti\n"); //stampa a video
39     printf ("Come posso aiutarti?\n"); //con caratteri speciali
40     printf ("\nA >> moltiplicare due numeri\n"); //\n
41     printf ("B >> Dividere due numeri\n"); //per le selezioni
42     printf ("C >> Inserire una stringa\n"); //del menu
43 }
44
45 void moltiplica ()    //funzione per l'operazione moltiplicazione
46 {
47     short int a;      //variabili int di tipo short equivalente a due byte
48     short int b;
49     short int prodotto;
50
51     printf ("Inserisci il primo fattore: ");
52     scanf ("%d", &a); //funzione per l'input da tastiera per inserire un valore numerico
53
54     printf ("\nInserisci il secondo fattore: ");
55     scanf ("%d", &b);
56
57     prodotto = a * b;
58
59     printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto);
60 }
61

```

Debugging Esercizio_5_Epicode_w2 (part. 2)

Analisi da Riga 30/33 a Riga 53/61

L'immagine a destra mostra il programma originale, l'immagine a sinistra il programma debuggato.

GNU nano 6.3 Esercizio_5_Epicode_w2.c *

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

GNU nano 6.3 Esercizio_5_Epicode_w2.c *

```
61
62 void dividi () //funzione per l'operazione divisione
63 {
64     int a; //variabili int
65     int b;
66     int divisione;
67
68     printf ("Inserisci il numeratore:");
69     scanf ("%d", &a);
70
71     printf ("\nInserisci il denominatore:");
72     scanf ("%d", &b);
73
74     divisione = a % b;
75
76     printf ("La divisione tra %d e %d e': %d", a,b,divisione);
77 }
78
79 void ins_string () //funzione per l'inserimento di una stringa
80 {
81     char stringa[10];
82     printf ("Inserisci la stringa: ");
83     scanf ("%s", &stringa);
84 }
85
```

Debugging Esercizio_5_Epicode_w2 (part. 3)

Analisi da Riga 53/61 a Riga 73/85

L'immagine a destra mostra il programma originale, l'immagine a sinistra il programma debuggato.