

Punto 1)

Il codice utilizzato serve a creare un programma che funge da assistente digitale ed offre all'utente 3 opzioni :

- due opzioni come se fosse una calcolatrice. Infatti svolge le operazioni di moltiplicazione o divisione di due numeri interi;
- un'opzione che permette di inserire una stringa.

Quando l'utente seleziona da un menu a scelta l'opzione desiderata, il programma svolge la relativa funzione e ne stampa il risultato.

PUNTO 2)

Le casistiche non gestite dal programma sono:

In void menu () : non viene calcolata l'eventualità di un inserimento sia di un carattere minuscolo anziché maiuscolo e di un carattere diverso da quelli elencati;

In main () nello switch non viene calcolato un default per gestire scelte non valide ;

In void moltiplica () : non viene considerata la casistica dell'inserimento di due numeri troppo alti e di numeri non interi; Non verifica se l'input sia effettivamente un numero, in caso contrario potrebbe generare un errore di formato di input

In void dividi () : non verifica se il denominatore è diverso da zero che comporterebbe un errore nella divisione; non verifica l'input e l'output di numeri razionali

In void ins_stringa () : Non verifica che l'input stringa non superi la lunghezza massima della stringa dichiarata.

In generale, per tutte le funzioni del codice non viene verificata la condizione in cui in input siano inserite le variabili richieste anziché altre.

Manca la libreria <string.h>

PUNTO 3)

Riporto di seguito il codice, evidenziando in rosso gli errori di sintassi e logici con accanto la correzione:

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

```
#include <string.h>
```

```
void menu ();
```

```
void moltiplica ();
```

```
void dividi ();
```

```
void ins_string();
```

```
int main ()
```

```
{
```

```
    char scelta = {'\0'};
```

```
    menu ();
```

```
    scanf ("%d", &scelta); %c
```

```
    switch (scelta)
```

```
    {
```

```
        case 'A':
```

```
            multiplica();
```

```
            break;
```

```
        case 'B':
```

```
            dividi();
```

```
            break;
```

```
        case 'C':
```

```
            ins_string();
```

```
            break;
```

```
    }
```

```
    return 0;
```

```
}
```

```
void menu ()
```

```
{
```

```
    printf ("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
```

```
    printf ("Come posso aiutarti?\n");
```

```
    printf ("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
```

```
}
```

```

void moltiplica ()
{
    short int a,b = 0; double
    printf ("Inserisci i due numeri da moltiplicare:");
    scanf ("%f", &a); %lf
    scanf ("%d", &b); %lf

    short int prodotto = a * b; double

    printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto); %lf per tutti
}

```

```

void dividi ()
{
    int a,b = 0; double
    printf ("Inserisci il numeratore:");
    scanf ("%d", &a); %lf
    printf ("Inserisci il denominatore:");
    scanf ("%d", &b); %lf

    int divisione = a % b; double ed al posto della% bisogna inserire \

    printf ("La divisione tra %d e %d e': %d", a,b,divisione); %lf per tutti
}

```

```
void ins_string ()
{
    char stringa[10];
    printf ("Inserisci la stringa:");
    scanf ("%s", &stringa); togliere la &
}
```

PUNTO 4

1) Inserire in main () precisamente alla fine del comando switch:

default:

```
printf("Scelta non valida, riprova.\n");
break;
```

per gestire il caso l'utente inserisca una risposta non valida;

Aggiungere il pipe minuscolo nella scelta delle lettere nel menù;

2) In void dividi () bisogna inserire una condizione in cui il denominatore inserito sia diverso da 0; per esempio:

```
if (b == 0) {
    printf("Il denominatore non può essere zero. Riprova.\n");
}
else {
    double divisione = a / b;
    printf("La divisione tra %lf e %lf è: %lf\n", a, b, divisione);
}
```

4) Per gestire il caso in cui l'utente inserisce una lettera o un carattere non numerico invece di un numero nella funzione "moltiplica ()" e "dividi ()", è possibile utilizzare la funzione scanf con il formato " %lf" e controllare il valore restituito dalla funzione scanf per verificare se l'input è valido, ed inserire tutto in un ciclo while. Esempio:

```
double a, b, result;
while (1) {
    printf("Inserisci i due numeri da moltiplicare:");
```

```

    if (scanf("%lf", &a) == 1 && scanf("%lf", &b) == 1) {
        break;
    }

    printf("Inserire solo numeri.\n");
    while (getchar() != '\n');
}

result = a * b;

printf("Il prodotto tra %lf e %lf e': %lf", a, b, result);
}

```

5) Per gestire il caso in cui vengano inseriti più di 10 caratteri all'interno della funzione string bisognerà creare un ciclo while in cui ogni volta l'utente inserisce più di 10 caratteri verrà visualizzato un messaggio di errore finché non soddisferà i requisiti richiesti.

Ad esempio:

```

void ins_string () {
    char stringa[10]; int length; do
    { printf ("Inserisci la stringa (massimo 10 caratteri):");
    fgets(stringa, sizeof(stringa), stdin);
    length = strlen(stringa) - 1; // -1 perché fgets include anche il carattere di nuova riga
    if (length > 10)
    { printf("La stringa inserita è troppo lunga, riprova\n"); }
    }
    while (length > 10);
}

```