

```

#include <stdio.h>

void menu ();
void moltiplica ();
void dividi ();
void ins_string();

int main ()
{
    char scelta = {'\0'};
    menu ();
    scanf ("%d", &scelta);

    switch (scelta)
    {
        case 'A':
            moltiplica();
            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;
    printf ("Inserisci i due numeri da moltiplicare:");
    scanf ("%f", &a);
    scanf ("%d", &b);

    short int prodotto = a * b;

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

```

```

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

    int divisione = a % b;

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

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

```

La funzione principale del programma: si tratta di un programma che può eseguire moltiplicazioni tra due numeri, eseguire divisioni tra due numeri e inserire una stringa.

Fra gli errori possiamo vedere:

1. Nello switch della funzione int **main**, troviamo un errore logico, si usa `scanf("%d", &scelta);` per leggere un carattere (**char**) si dovrebbe utilizzare `%c` anziché `%d`.
2. Nella funzione moltiplica viene usato `"%f"` per leggere il primo numero, ma è stato dichiarato "a, b" come short int. Si dovrebbe usare `%hd` anziché `%f` per leggere un intero corto. Inoltre nella funzione non viene gestita la possibilità di overflow quando si calcola il prodotto. Se i numeri inseriti sono troppo grandi, si otterrebbero risultati inaspettati.
3. Nella funzione moltiplica inoltre viene inizializzato solo "b", dobbiamo inizializzare anche a. Quindi: `short int a = 0, b = 0;`
4. Nella funzione `ins_string`, viene usato `scanf("%s", &stringa);` ma `stringa` è un array di caratteri, quindi non è necessario utilizzare `&` con `%s`.
5. Valori non validi per la scelta del menu:
Errore di carattere logico; se l'utente inserisce un carattere diverso da 'A', 'B' o 'C', il programma non gestisce questa situazione. Si dovrebbe considerare l'aggiunta di un default per gestire input non validi. (Es: default:

```

printf("Non posso elaborare la tua richiesta");
break;

```
6. Divisione per zero:
Errore logico. Nel caso in cui l'utente selezioni la divisione ('B'), il programma non gestisce la possibilità di inserire un denominatore uguale a zero, che porterebbe a una divisione per zero e comporterebbe un comportamento indefinito. Dobbiamo specificare:

```

if (b != 0) {
    int divisione = a / b;
    printf("La divisione tra %d e %d e': %d\n", a, b, divisione);
} else {
    printf("Impossibile dividere per zero.\n");
}

```

Ovviamente anche il segno della percentuale va tolto, perché qui vogliamo che il programma esegua la divisione non il resto.
7. Gestione della stringa:
Overflow > un errore nella rappresentazione di un certo numero dovuto al fatto che la quantità di cifre disponibili è minore rispetto a quelle necessarie per rappresentarlo. Nella funzione `ins_string()`, il programma può avere problemi se l'utente inserisce una stringa più lunga dei caratteri indicati, perché la dimensione dell'array è limitata, andrebbe modificato per consentire l'inserimento di stringhe più lunghe > `char stringa [100];` //o qualsiasi dimensione appropriata.

8. Inoltre non c'è la possibilità, come invece abbiamo inserito per il gioco quiz elaborato in precedenza, di poter ripetere il tutto. Ovviamente questo ricade poi nella scelta di cosa fa il programma, e anche di user experience.