

## Bug Hunting

Analizziamo il codice dato in consegna per cercare possibili errori e per cercare di ottimizzarlo al meglio andando anche ad inserire dei commenti per rendere più semplice la lettura del codice:

```
GNU nano 6.3 Esercizio_10_Epicode.c *
#include <stdio.h> //Ai fini dell'esecuzione dell'eseguibile, è consigliato mettere la libreria <stdlib.h> per inserire la funzione system("pause")
//mentre è obbligatorio inserire le librerie <stdbool.h> e <string.h> per ottimizzare al meglio i controlli

void menu (); //Dichiarazione della funzione menu() di tipo void, ossia alla fine della funzione, non restituisce nessun valore
void moltiplica (); //Dichiarazione della funzione moltiplica () di tipo void
void dividi (); //Dichiarazione della funzione dividi() di tipo void
void ins_string(); //Dichiarazione della funzione ins_string() di tipo void

int main () //Chiamata della funzione principale main() di tipo int
{
    char scelta = {'\0'}; //Dichiarazione della variabile scelta di tipo char ed inizializzazione a NULL
    menu (); //Chiamata della funzione menu()
    scanf ("%d", &scelta); //Inserimento della scelta data dal menù errata poiché la variabile è stata dichiarata come carattere mentre nello scanf viene inserita come un intero

    switch (scelta) //Utilizzo della funzione switch per gestire la casistica della scelta senza andare a occuparsi di possibile errore dell'utente
    {
        case 'A': //In caso di input dell'utente del carattere A, esegue la funzione moltiplica()
            moltiplica();
            break; //Comando per uscire dallo switch evitare la lettura dagli altri case
        case 'B': //In caso di input dell'utente del carattere B, esegue la funzione dividi()
            dividi();
            break;
        case 'C': //In caso di input dell'utente del carattere C, esegue la funzione ins_string()
            ins_string();
            break;
    }

    return 0; //Restituisce valore 0 alla funzione main e termina il programma. Per pura scelta grafica e per una lettura migliore, è consigliabile indentarlo
}
```

```
void menu () //Questa funzione menu() serve per stampare 3 stringe in modo tale che l'utente possa avere un'interfaccia.
//Consigliabile aggiungere una 4 scelta in cui viene data l'opzione di uscire dal programma
{
    printf ("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n"); //Correggere assistente in assistente
    printf ("Come posso aiutarti?\n");
    printf ("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
}

void moltiplica () //Questa funzione moltiplica () serve a moltiplicare 2 numeri dati in input.
{
    short int a,b = 0; //Dichiara 2 variabili, a e b, di tipo short inizializzando b a 0. Correggerei dichiarando semplicemente a e b come 2 variabili int
    printf ("Inserisci i due numeri da moltiplicare:");
    scanf ("%f", &a); //Inserimento errato del primo numero poiché %f fa riferimento alle variabili di tipo float
    scanf ("%d", &b); //Inserimento errato del secondo numero poiché %d fa riferimento alle variabili di tipo int
    //Aggiungerei 2 printf per facilitare la comprensione dell'inserimento dei numeri
    //Dichiarando a e b come 2 variabili int, dentro gli scanf andrei ad inserire %d

    short int prodotto = a * b; //Dichiarazione di tipo int ed assegnazione di prodotto alla moltiplicazione tra a e b. Correggerei dichiarandolo int

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

```
void dividi () //Questa funzione dividi() serve a dividere 2 numeri dati in input
{
    int a,b = 0; //Dichiarazione di due variabili intere inizializzando b a 0.
    printf ("Inserisci il numeratore:");
    scanf ("%d", &a);
    printf ("Inserisci il denominatore:"); //Stampa errata poiché si scrive denominatore
    scanf ("%d", &b);

    int divisione = a % b; //Dichiarazione ed assegnazione completamente sbagliata poiché dichiara la variabile divisione come un intero, senza considerare una possibile soluzione con la virgola
    //e soprattutto l'assegnazione è sbagliata poiché non fa la divisione la il modulo tra a e b, il quale serve a calcolare il RESTO di una funzione.
    //Per correggerlo si deve dichiarare divisione come un float e scrivere l'assegnazione nel seguente modo: a/b;

    printf ("La divisione tra %d e %d e': %d", a,b,divisione); //Di conseguenza, la stampa della divisione deve essere cambiata da %d a %f
}

void ins_string () //Questa funzione ins_string() serve per l'inserimento di una stringa di grandezza 10
{
    char stringa[10];
    printf ("Inserisci la stringa:");
    scanf ("%s", &stringa);
}
```

Adesso andiamo ad applicare diverse modifiche in modo tale da controllare la casistica dello switch ed ottimizzare il codice al meglio:

```

GNU nano 6.3                                     Esercizio_10_Epicode_Corretto.c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdbool.h>

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

int main ()
{
    char scelta_iniziale = {'\0'};
    char scelta_finale = {'\0'};
    bool s;
    do{
        menu ();
        do{
            scanf ("%c", &scelta_iniziale);
            rewind(stdin);
            if(scelta_iniziale=='a')
                scelta_iniziale='A';
            if(scelta_iniziale=='b')
                scelta_iniziale='B';
            if(scelta_iniziale=='c')
                scelta_iniziale='C';
            if(scelta_iniziale=='d')
                scelta_iniziale='D';
            switch (scelta_iniziale)
            {
                case 'A':{
                    moltiplica();
                    s=false;
                    break;

```

```

                case 'B':{
                    dividi();
                    s=false;
                    break;
                }
                case 'C':{
                    ins_string();
                    s=false;
                    break;
                }
                case 'D':
                {
                    printf("\nGrazie ed alla prossima\n");
                    system("pause");
                    return 0;
                }
                default:{
                    printf("\nHai sbagliato ad inserire la scelta, riprova: ");
                    s=true;
                    break;
                }
            }
        }while(s);
        printf("\nCiao, vuoi continuare ad usare i miei servizi?\n\nRispondi Y oppure clicca qualsiasi altro tasto per uscire: ");
        scanf ("%c", &scelta_finale);
        rewind(stdin);
    }while(scelta_finale!='Y' || scelta_finale=='y');
    printf("\nGrazie per avermi usato\n\n");
    system("pause");
    return 0;
}

```

In aggiunta al codice precedente, sono andato ad importare 2 librerie ossia <string.h> e <stdbool.h> e sono andato a creare 2 do-while. Il primo serve per far ripetere il programma oppure uscirne fuori tramite **scelta\_finale** mentre il secondo per il controllo della scelta iniziale. Sono andato ad aggiungere delle condizioni in modo tale che se l'utente inserisca una delle 4 possibilità ma con la lettera minuscola, vado ad impostarla maiuscola, così da poter entrare nello switch-case. Una piccola aggiunta è stata quella della libreria <stdlib.h> per poter aggiungere il comando system("pause") in modo da non far terminare subito il programma.

Ora passiamo alle funzioni:

```

GNU nano 6.3                                     Esercizio_10_Epicode_Corretto.c
void menu ()
{
    printf ("\nBenvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
    printf ("\nCome posso aiutarti?\n");
    printf ("\nA >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\nD >> Esci\n\nRisposta: ");
}

void moltiplica ()
{
    int a, b, prodotto;
    printf ("\nInserisci i due numeri da moltiplicare\n");
    printf ("\nPrimo fattore: ");
    scanf ("%d", &a);
    rewind(stdin);
    printf ("\nSecondo fattore: ");
    scanf ("%d", &b);
    rewind(stdin);
    prodotto = a * b;
    printf ("\nIl prodotto tra %d e %d e': %d\n\n", a, b, prodotto);
}

```

```

void dividi ()
{
    int a,b;
    printf("\nInserisci il numeratore: ");
    scanf ("%d", &a);
    rewind(stdin);
    do{
        printf("\nInserisci il denominatore: ");
        scanf ("%d", &b);
        rewind(stdin);
        if(b==0)
            printf("\nNon puoi dividere un numero per 0!!\n");
        }while(b==0);

    float divisione = (float)a / (float)b;

    printf ("\nLa divisione tra %d e %d e': %f\n\n", a,b,divisione);
}

void ins_string ()
{
    char stringa[10];
    int l;
    printf ("\nInserisci la stringa: ");
    fgets (stringa, 10 ,stdin);
    stringa[strlen(stringa)-1] = '\0';
    printf("\n\nQuesta e' la tua stringa: %s\n", stringa);
}

```

Come si può notare, alla funzione `menu ()` sono andato ad aggiungere una quarta opzione, **D>> Esci** in modo tale che l'utente possa scegliere di uscire subito dal programma. Nella funzione `moltiplica()` sono andato a cambiare i valori da short int ad int per avere un intervallo di valori maggiore, poi a migliorarne l'estetica. Nella funzione `dividi()` sono andato a fare un controllo sul denominatore, in modo tale che l'utente non possa inserire 0, poi ho cambiato il tipo della divisione da int a float, ho castato a e b in modo tale da renderli float e cosa più importante, sono andato a sostituire il modulo `%` con `/`. Infine sono andato a fare un leggero controllo alla funzione `ins_string()` in modo tale da non permettere di andare oltre al numero consentito di caratteri e poi in aggiunta una stampa per controllare se la stringa venisse inserita correttamente.