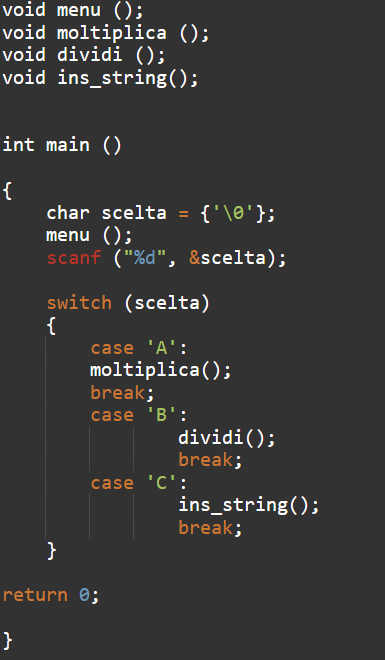
# **ESERCITAZIONE C**

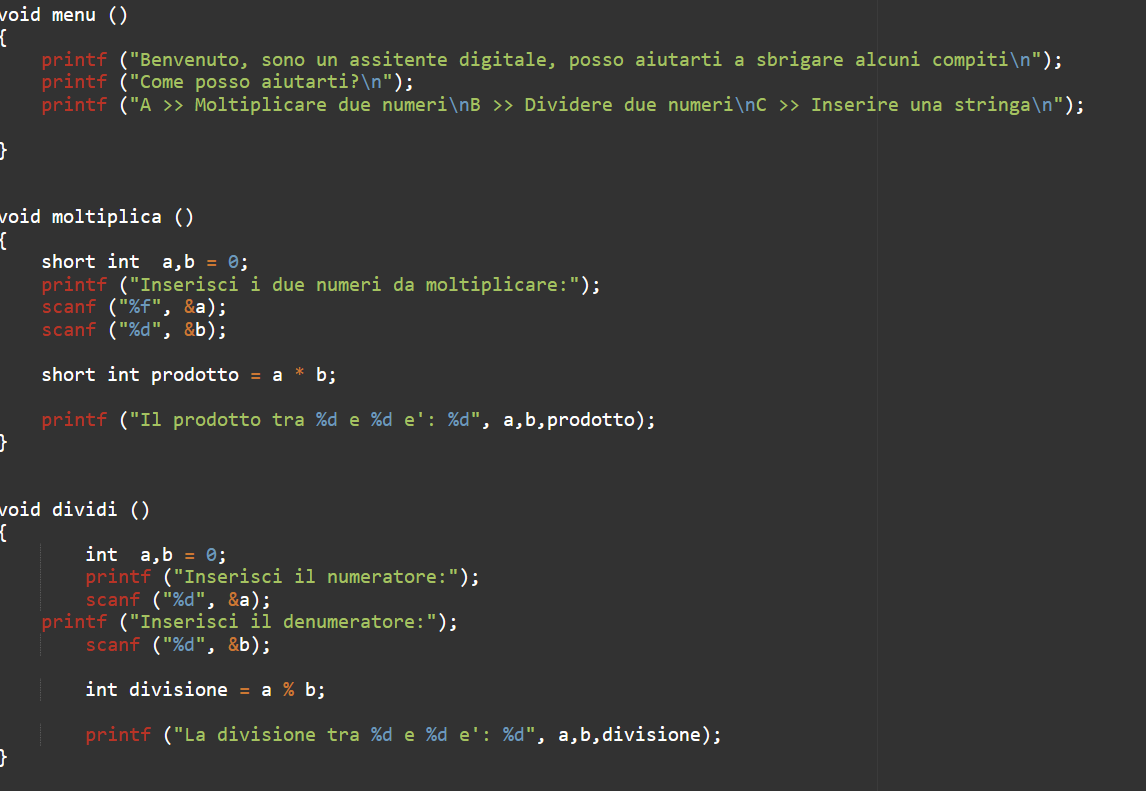
Dato il seguente codice avevo il compito di trovare eventuali errori



Il codice presentava diversi errori di sintassi e di progettazione ;

Per prima cosa ho corretto gli errori di sintassi;

Qui è pressentè il primo errore alla variabile scelta che è una stringa è stato assegnato il puntatore %d il quale va bene solo per variabili int, mentre per le variabili char va assegnato il puntatore %s

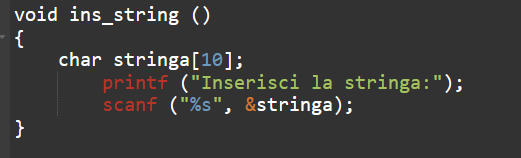


# Funzione moltiplica errori sintassi

.1)Il primo errore trovato e nella funzione moltiplica infatti ho cambiato il parametro short int per le variabili a e b in int, poiché short int permette di inserire un range di valori compresi –32,768 e 32,768, mentre int puo consentire di inserire valori piu alti quindi preferibile per operazioni di moltiplicazione,.2) secondo errore alla variabile A viene assegnato float mentre nella dichiarazione è interro quindi o assegnato il valore corretto %d, .3) il terzo errore e il parametro short int nella variabile prodotto come gia indicato nel .1)

# Funzione dividi errori sintassi

che la divisione pr.1 primo errore trovato e la dichiarazione di a e b come variabili int e assegnazione di puntatore %d io ho modificato in variabili float , e anche la variabile divisione poi evede spesso risultati con la virgola la modifica è stata fatta anche nella restituzione del valore.



Qui invece non sono stati induati errori di sintassi

# Errori di progettazione

Bisogna fare molta attenzione a errori di progettazione poiché potrebbero portare a bug nel codice che un un black hat potrebbe sfruttare a suo favore.sono partito ridisgnado un diagramma di flusso migliore .

Inizio

V

Esegui menu()

|

V

Richiedi input utente (scelta)

|

V

|

+---A---> Esegui moltiplica()

| |

| V

| Torna al menu principale

|

+---B---> Esegui dividi()

| |

| V

| Torna al menu principale

+---C---> Esegui ins\_string()

| |

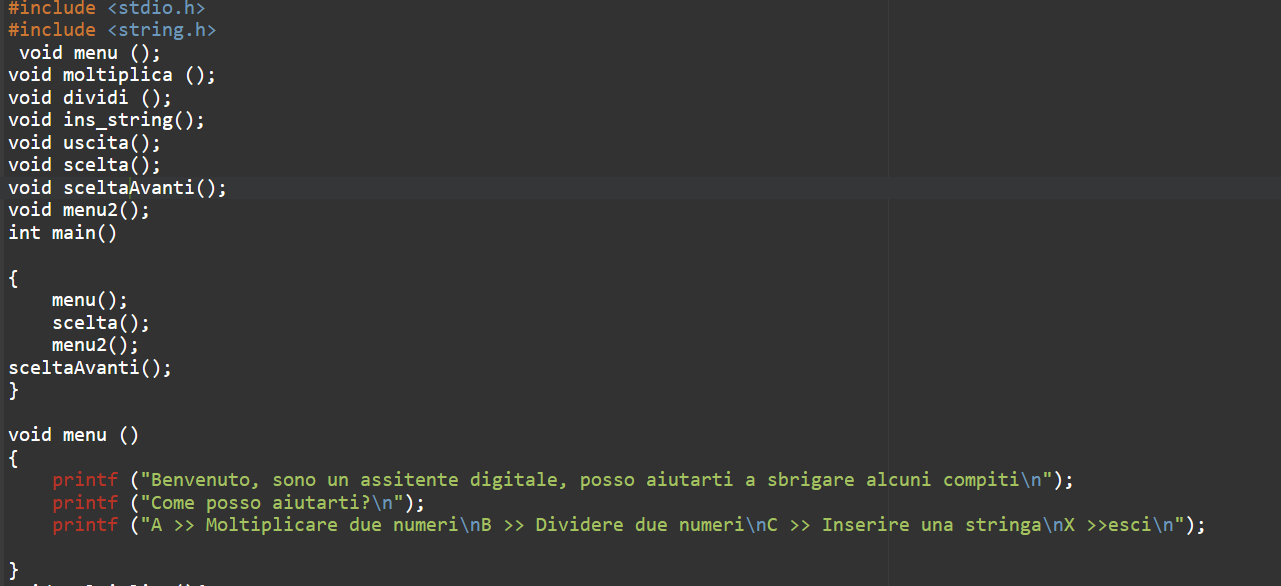
| V

| Torna al menu principale

+---X---> Esegui uscita()

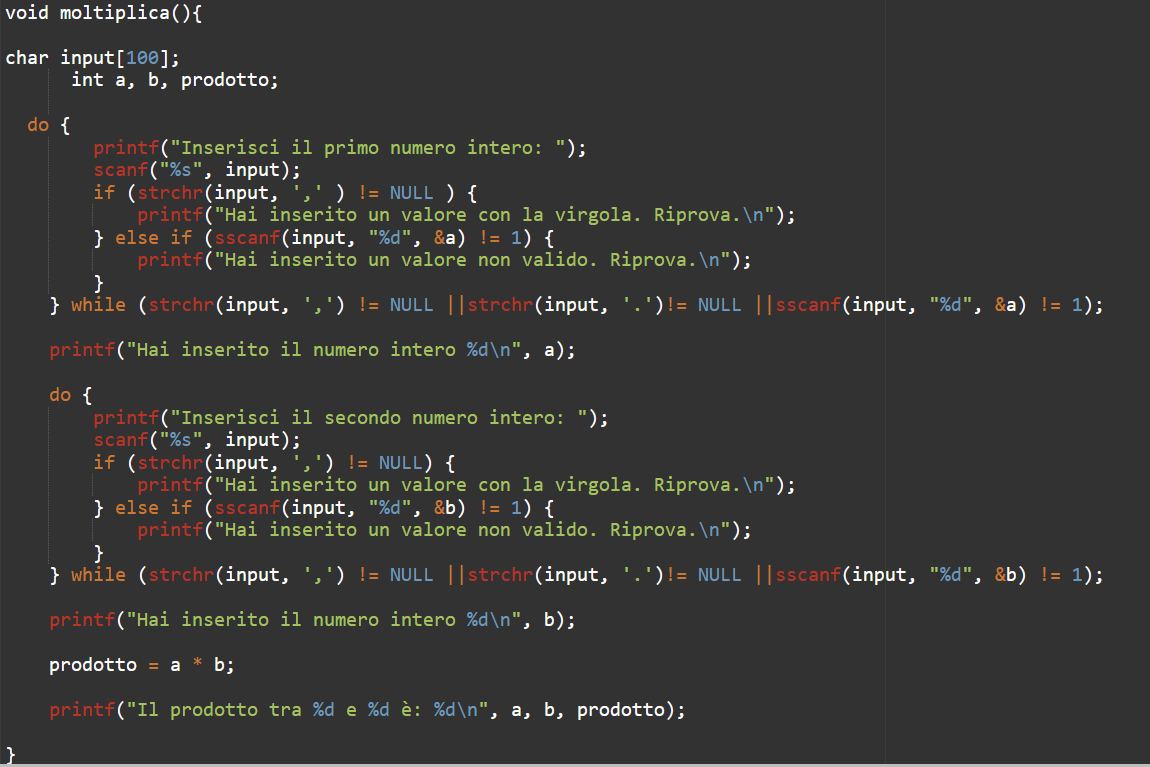
| termina programma

# Prime differrenze

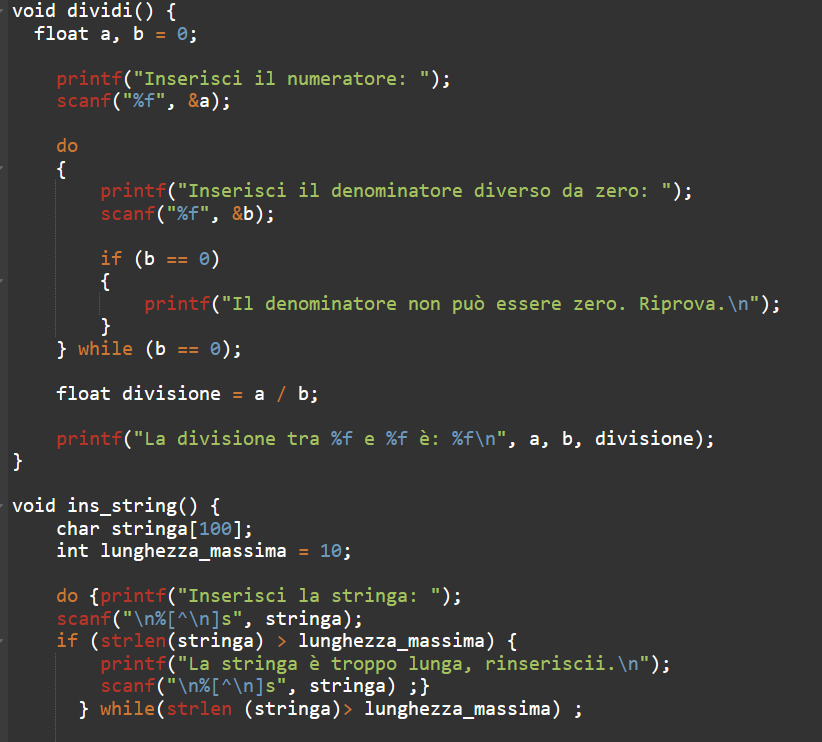


Ho incluso la libreria <string.h> che una libreria molto utile per manipolare le stringhe più tardi vedremo come l'ho usata,

Ho creato altre funzioni non esistenti prima come la funzione uscita, che ci permetterà di uscire dal programma, ho trasformato la scelta presente precedentemente in una funzione, poi ho creato la funzione menu2 la quale ci permetterà far scegliere all'utente se uscire dal programma o continuare con altre operazioni, ho creato la funzione scelta avanti la quale eseguirà ciò che richiesto dall'utente, quindi di conseguenza modificato il main con le nuove funzioni . HO modificato anche la funione menu() inserendo do la possibilità di uscita con x.



Qui ho fatto la prima grande modifica nella funzione moltiplica, poiché come era scritta precedentemente nel codice l'utente poteva inserire numeri decimali e la funzione non l avrebbe considerati, creando un possibile problema. Poiché, non dava nessun errore di immissione; e in più ho considerato che l'operazione fosse concessa solo con numeri interi, nel caso contrario sarebbe bastato cambiare le variabili in float. Quindi ho creato una nuova variabile di tipo Char chiamata input, questo per poter analizzare se l'utente mettesse (.)O (; )nel numero da lui inserito con la funzione strchr la quale ha il compito di trovare tali lettere nella stringa, poi ho detto al proggramma in caso di valori corrispondenti a . O ; poi di controllare se il numero inserito sia un intero, di fermarsi e dare a l utente la possibilità di rinserire finché entrambi i parametri non sono giusti, questo sia per a e sia per b, poi di restuire il risultato della moltiplicazione.

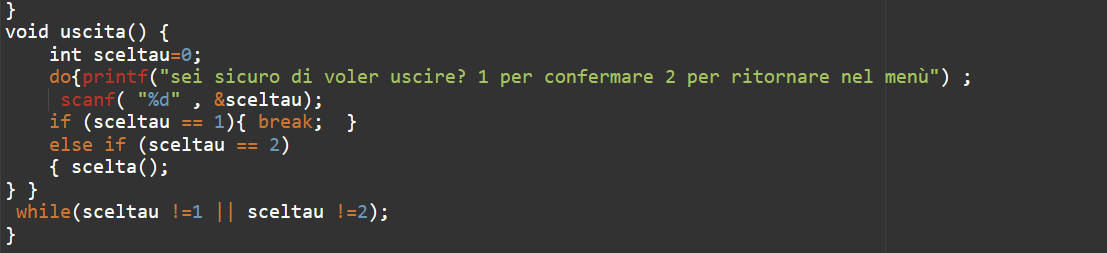


L'unica modifica fatta alla funzione dividi, è quella di non permettere le divisioni per 0 in quanto matematicamente impossibili,

quindi di ridare l'errore finché il denominatore non sarà diverso da 0 e poi restituire il risultato della divisione.

La modifica che ho fatto nella funzione inserisci stringa e che l'utente non potesse inserire stringe troppo lunghe per evitare bug; quindi, ho creato un nuovo valore lunghezza massima, il quale sarà usato come valore di riferimento per il confronto con la stringa tramite la funzione strlen ,

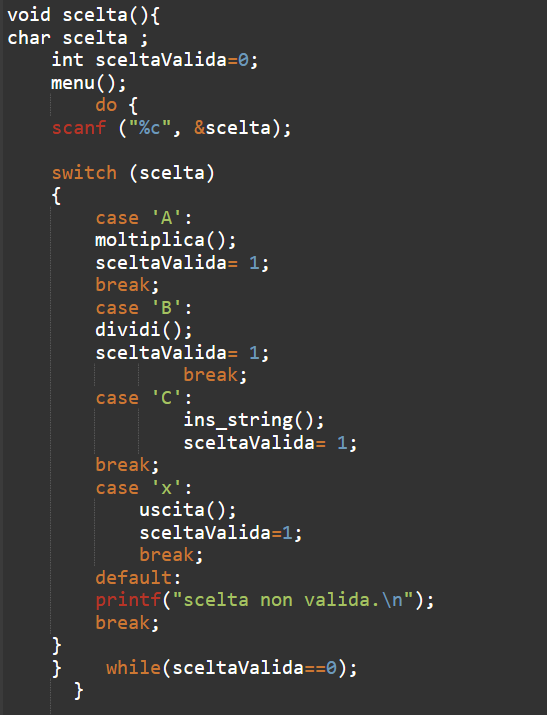
E fara reinserire la stringa finché non ci sarà una stringa pari o minore a 10

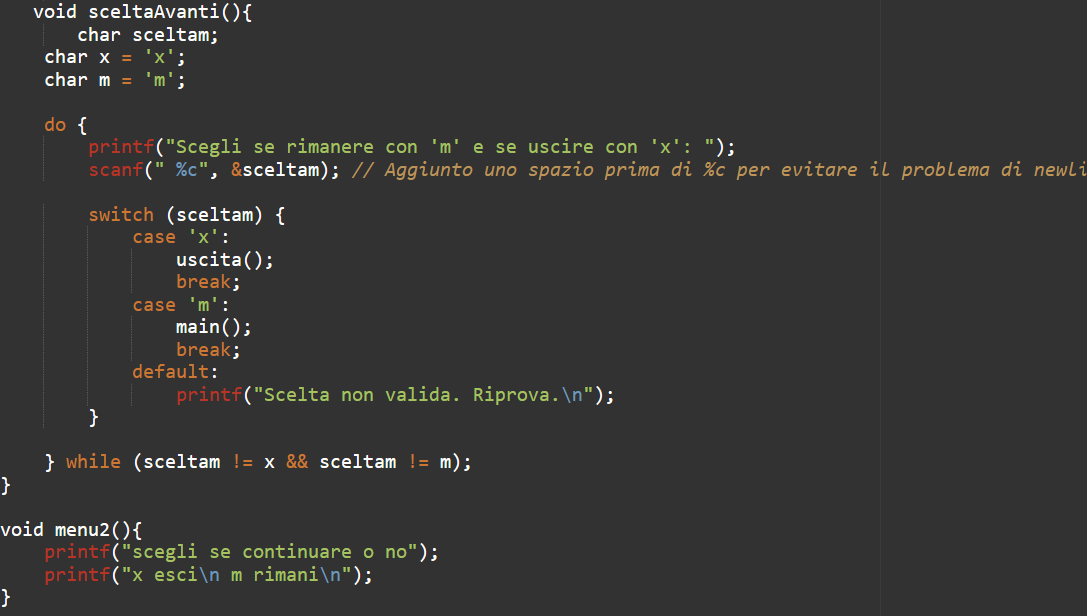


Qui ho creato la funzione uscita che darà la possibilità all'utente di uscire,

Avrà 2 opzioni dati da volere 1 e 2 in caso di valore 1 uscirà ,in caso di valore 2

Ritornera a scegliere quale operazione fare, tale ciclo continuerà finché non si inserisce uno dei due valori prestabiliti.



Qui ho trasformo “scelta” precedente presente nel main in una funzione il problema che aveva era il fatto di poter inserire qualsiasi lettera e per lettere errate il programma si chiudeva, per ovviare a questo ho dichiarato una variabile chiamata int valore scelta assegnandole valore 0, e modificandola per ogni caso corretto, cosi il programma riinizia il ciclo dando messaggio di errore finché valore scelta non è modificato.

Poi ho creato queste 2 funzioni per evitare che l utente uscisse involontariamente dal processo chiedendoli la conferma o meno e in caso di errore di rientrare nel main .