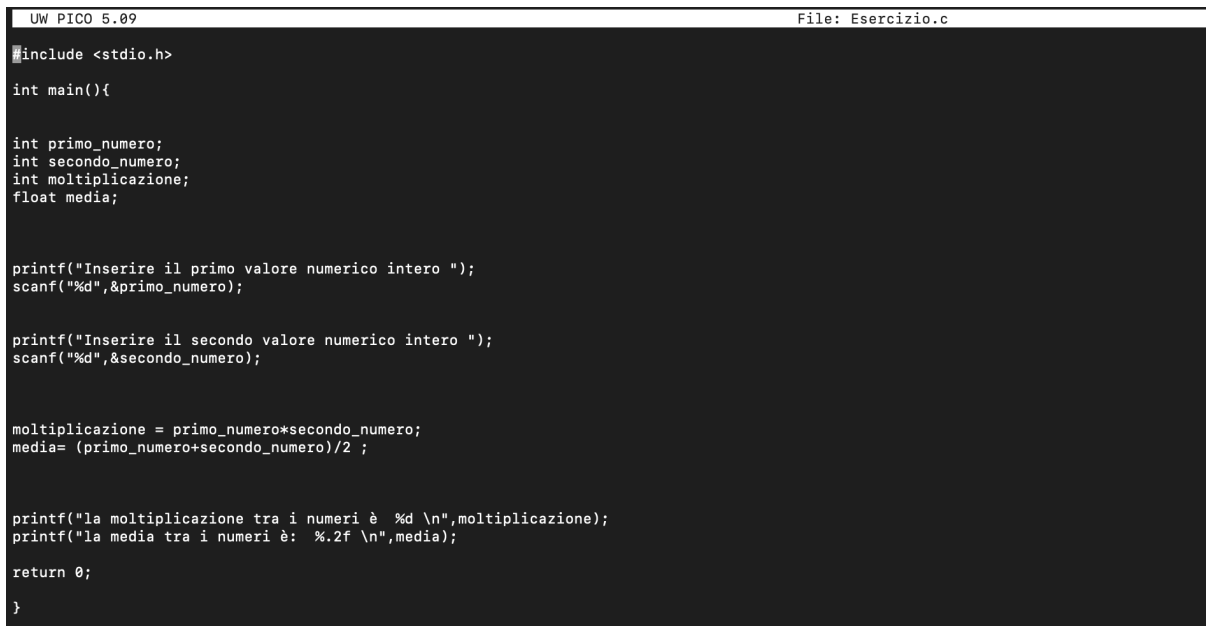


Lezione S2/L2

Linguaggio C

La Lezione di oggi ha richiesto lo sviluppo di un programma utilizzando il linguaggio di programmazione C. in questo programma dovevamo:

- Creare un programma che esegua una moltiplicazione tra numeri inseriti dall'utente.
- Creare un programma che esegua la media tra numeri inseriti dall'utente.

The image shows a screenshot of a code editor window. The title bar at the top reads 'UW PICO 5.09' on the left and 'File: Esercizio.c' on the right. The editor area has a dark background with light-colored text. The code is as follows:

```
#include <stdio.h>

int main(){

    int primo_numero;
    int secondo_numero;
    int moltiplicazione;
    float media;

    printf("Inserire il primo valore numerico intero ");
    scanf("%d",&primo_numero);

    printf("Inserire il secondo valore numerico intero ");
    scanf("%d",&secondo_numero);

    moltiplicazione = primo_numero*secondo_numero;
    media= (primo_numero+secondo_numero)/2 ;

    printf("la moltiplicazione tra i numeri è %d \n",moltiplicazione);
    printf("la media tra i numeri è: %.2f \n",media);

    return 0;
}
```

figura 1: Il codice

Analizziamo ora le varie parti del codice e capiamo come siamo arrivati a questo risultato. Personalmente ho svolto il programma su terminale mac, i comandi sono praticamente uguali e il procedimento è esattamente uguale.

Innanzitutto dobbiamo creare il file "Esercizio.c" per fare questo apriamo il terminale, viaggiamo nella directory dove vogliamo lavorare ed utilizziamo il comando:

nano Esercizio.c

questo comando crea e permette di editare il file, (in caso si può usare anche il comando **touch** per creare il file).

il codice si divide in 4 parti (figura 2):

INIZIALIZZAZIONE : in questa parte inizializziamo le variabili che useremo nel nostro codice, molto importante stabilire la tipologia int o float delle variabili. Solitamente le variabili vengono poste tutte = 0 manualmente perchè sennò posseggono un valore imprevedibile, nel nostro caso non è necessario perchè vengono modificate nel codice ma è buona prassi azzerare le variabili.

INPUT CLI: qui abbiamo la parte di codice che richiede all'utente di inserire i numeri per poter svolgere la moltiplicazione e la media, importante notare come nella funzione **scanf** abbiamo inserito il termine “%d” per indicare un valore intero e “&primo_numero” per indicare l'indirizzo di quella variabile.

CALCOLI: senza necessità di usare librerie esterne per il calcolo matematico svolgiamo le operazioni assegnando i valori alle variabili moltiplicazione e media, ovviamente ci assicuriamo che media sia un valore **float**

OUTPUT: stampa sul terminale il risultato della moltiplicazione e la media calcolata.

```
UW PICO 5.09 File: Esercizio.c
#include <stdio.h>
int main(){
    int primo_numero;
    int secondo_numero;
    int moltiplicazione;
    float media;

    printf("Inserire il primo valore numerico intero ");
    scanf("%d",&primo_numero);

    printf("Inserire il secondo valore numerico intero ");
    scanf("%d",&secondo_numero);

    moltiplicazione = primo_numero*secondo_numero;
    media= (primo_numero+secondo_numero)/2 ;

    printf("la moltiplicazione tra i numeri è %d \n",moltiplicazione);
    printf("la media tra i numeri è: %.2f \n",media);
    return 0;
}
```

The code is annotated with four colored boxes and labels:

- INIZIALIZZAZIONE** (yellow box):
`int primo_numero;`
`int secondo_numero;`
`int moltiplicazione;`
`float media;`
- INPUT CLI** (red box):
`printf("Inserire il primo valore numerico intero ");`
`scanf("%d",&primo_numero);`
`printf("Inserire il secondo valore numerico intero ");`
`scanf("%d",&secondo_numero);`
- CALCOLI** (green box):
`moltiplicazione = primo_numero*secondo_numero;`
`media= (primo_numero+secondo_numero)/2 ;`
- OUTPUT** (purple box):
`printf("la moltiplicazione tra i numeri è %d \n",moltiplicazione);`
`printf("la media tra i numeri è: %.2f \n",media);`
`return 0;`

figura 2: breakdown del codice

Per utilizzare il programma dobbiamo prima salvarlo con “**ctrl + O**” ed uscire con “**ctrl + X**”, nella CLI digitiamo il comando **gcc -o Esercizio Esercizio.c** , che crea l'eseguibile **Esercizio**, seguito dal comando **./Esercizio** per eseguirlo, i risultati sono osservabili in figura 3.

```
(base) matteobosio@MacBookdiMatteo Desktop % nano Esercizio.c
(base) matteobosio@MacBookdiMatteo Desktop % gcc -o Esercizio Esercizio.c
(base) matteobosio@MacBookdiMatteo Desktop % ./Esercizio
Inserire il primo valore numerico intero 10
Inserire il secondo valore numerico intero 340
la moltiplicazione tra i numeri è 3400
la media tra i numeri è: 175.00
(base) matteobosio@MacBookdiMatteo Desktop % ./Esercizio
Inserire il primo valore numerico intero 99
Inserire il secondo valore numerico intero 32
la moltiplicazione tra i numeri è 3168
la media tra i numeri è: 65.00
```

figura 3: Esecuzione Programma

Utilizzando il programma ho notato che non ci sono condizioni per evitare che l'utente inserisca un numero casuale o un carattere char , ho così deciso di implementare un controllo che crea un loop dal quale l'utente può uscire solo se inserisce il corretto valore. Ecco il codice:

```
#include <stdio.h>

int main(){

int primo_numero;
int secondo_numero;
int moltiplicazione;
float media;

int valid;
int valid2;

do { printf("Inserire il primo valore numerico intero: ");
valid = scanf("%d", &primo_numero);

if (valid != 1) {
printf("Errore: Il valore inserito non è un intero. Riprova.\n");

while (getchar() != '\n');
} while (valid != 1);

do { printf("Inserire il secondo valore numerico intero: ");
valid2 = scanf("%d", &secondo_numero);

if (valid2 != 1) {
printf("Errore: Il valore inserito non è un intero. Riprova.\n");

while (getchar() != '\n');
} while (valid2 != 1);

moltiplicazione = primo_numero*secondo_numero;
media= (primo_numero+secondo_numero)/2.0 ;

printf("la moltiplicazione tra i numeri è %d \n",moltiplicazione);
printf("la media tra i numeri è: %.2f \n",media);

return 0;
}
```

IMPLEMENTAZIONE DI CHECK

figura 4: implementazione di sicurezza.

Questa implementazione inizia con la creazione di due variabili int, queste variabili vengono usate nel ciclo **do-while**, come vediamo assegniamo a valid il valore di scanf , ma cosa significa?.

Praticamente valid assumerà un valore 0 (in caso di input errato dell'utente) oppure un valore 1 (in caso di valore corretto), se il valore è errato allora viene stampata la linea di errore e viene richiesto di nuovo l'input. Notiamo anche la riga con **getchar()** questa serve per pulire il valore di input scritto e permette la scrittura di un nuovo valore.

Ripetiamo il processo per entrambi i numeri ed otteniamo il seguente risultato:

```
((base) matteobosio@MacBookdiMatteo Desktop % gcc -o Esercizio Esercizio.c
((base) matteobosio@MacBookdiMatteo Desktop % ./Esercizio
Inserire il primo valore numerico intero: a
Errore: Il valore inserito non è un intero. Riprova.
Inserire il primo valore numerico intero: à
Errore: Il valore inserito non è un intero. Riprova.
Inserire il primo valore numerico intero: .7
Errore: Il valore inserito non è un intero. Riprova.
Inserire il primo valore numerico intero: 8
Inserire il secondo valore numerico intero: b
Errore: Il valore inserito non è un intero. Riprova.
Inserire il secondo valore numerico intero: +
Errore: Il valore inserito non è un intero. Riprova.
Inserire il secondo valore numerico intero: $
Errore: Il valore inserito non è un intero. Riprova.
Inserire il secondo valore numerico intero: 9
la moltiplicazione tra i numeri è 72
la media tra i numeri è: 8.50
```

figura 5: codice con check

Purtroppo il programma presenta comunque delle limitazioni quando inseriamo dei numeri float. Questo perché il sistema riconosce che il numero è errato ma passa solo il termine intero del float e quindi viene poi preso come buono ecco qua un esempio:

```
((base) matteobosio@MacBookdiMatteo Desktop % ./Esercizio
Inserire il primo valore numerico intero: 7.6
Inserire il secondo valore numerico intero: Errore: Il valore inserito non è un intero. Riprova.
Inserire il secondo valore numerico intero: 6.9
la moltiplicazione tra i numeri è 42
la media tra i numeri è: 6.50
```

figura 6: il Problema