C - ES01 - LA SELEZIONE

Tommaso Sollo 3°H 26/10/2022

## 0 - Introduzione preliminare

## 0.1 - Scopo del documento

Lo scopo del presente documento è quello di verificare il funzionamento dei programmi realizzati tramite collaudi.

# Esercizio a

## Progetto

Scrivere un programma che, richiesto un numero intero, visualizzi tutti i suoi divisori.

## 

## Implementazione

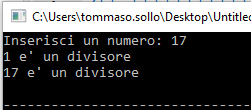
[Link al file in c](https://github.com/tommasosollo/INFORMATICA1/blob/main/ES02%20-%20Cicli%20e%20iterazioni/a_divisoreNumero.c)

## Collaudo

Inserendo 18, l’output desiderato dovrebbero essere i seguenti numeri: 2, 3, 6, 9.

### 

Inserendo 17, che è un numero primo, l’output desiderato è 1 e il numero stesso



# Esercizio b

## Progetto

Scrivere un programma che visualizzi tutte le coppie ordinate di numeri naturali la cui somma è 15.

## 

## Implementazione

[Link al file in c](https://github.com/tommasosollo/INFORMATICA1/blob/main/ES02%20-%20Cicli%20e%20iterazioni/b_coppie_somma_15.c)

## Collaudo

### 

# Esercizio c

## Progetto

Si scriva un programma per calcolare la media aritmetica di una serie di numeri inseriti da tastiera. L’introduzione del valore 0 indica il termine del caricamento dei dati.

## 

## Implementazione

[Link al file in c](https://github.com/tommasosollo/INFORMATICA1/blob/main/ES02%20-%20Cicli%20e%20iterazioni/c_mediaNumeri.c)

## Collaudo

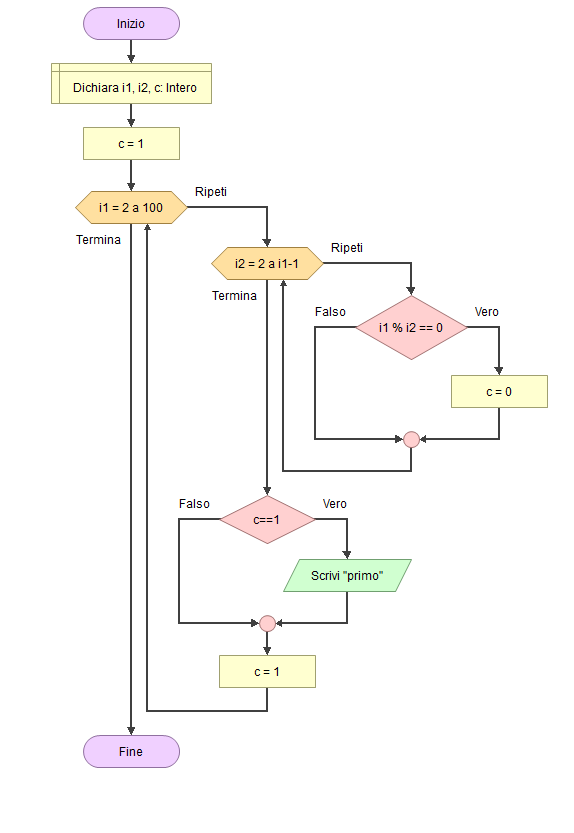
### 

Il programma funziona correttamente, infatti inserendo i numeri 2, 3 e 4, la cui media è 3, il programma visualizza correttamente il risultato.

# Esercizio d

## Progetto

Scrivere un programma che stampi tutti i numeri primi compresi tra 1 e 100. Un numero è primo quando ha come divisore uno e sé stesso. Quindi è primo ciascun numero naturale maggiore di 1 che sia divisibile solamente per 1 e per sé stesso.



## Implementazione

[Link al file in c](https://github.com/tommasosollo/INFORMATICA1/blob/main/ES02%20-%20Cicli%20e%20iterazioni/d_numeriPrimi100.c)

## Collaudo

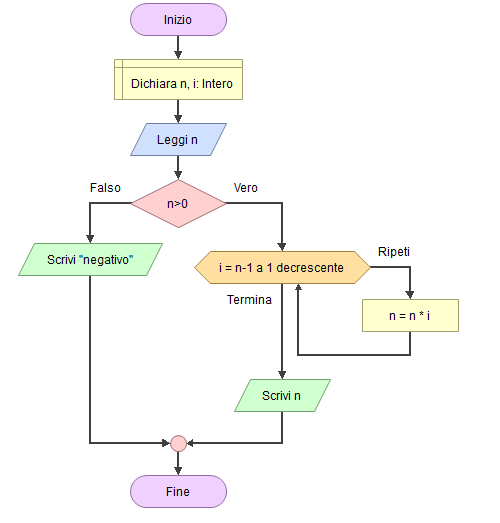
### 

Il programma stampa correttamente tutti i numeri primi dal 1 al 100, estremi esclusi.

# Esercizio e

## Progetto

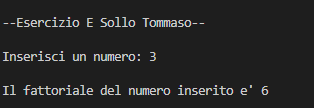
Si scriva un programma che calcoli il fattoriale di un numero intero N dato dalla tastiera. Si ricordi che il fattoriale di un numero n (simbolo n!) viene calcolato con la seguente formula: n! = n ·(n–1)·(n–2)· ... ·2 ·1.



## Implementazione

[Link al file in c](https://github.com/tommasosollo/INFORMATICA1/blob/main/ES02%20-%20Cicli%20e%20iterazioni/e_fattorialeNumero.c)

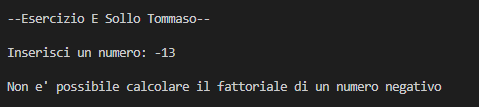
## Collaudo



Il programma funziona correttamente, infatti inserendo il numero 3, il cui fattoriale è uguale a 3! = 3 \* 2 \* 1 = 6, il risultato è 6.

### 

Si nota che il valore massimo inserito per cui il risultato rimane corretto e’ 12, dopo questa soglia il programma va in overflow



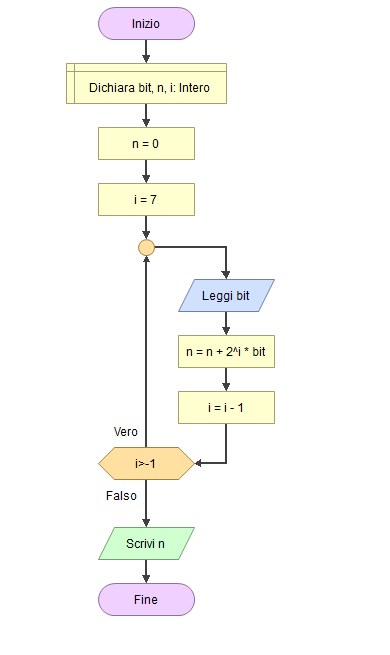
Se il valore inserito è minore di 0, il programma non esegue il calcolo in quanto non esiste il fattoriale di un numero negativo.

# 

# Esercizio f

## Progetto

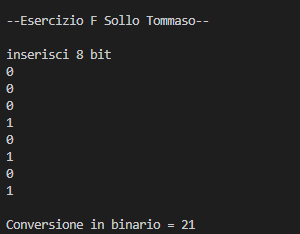
Scrivere un programma che converte un numero binario a 8 bit in un numero decimale. L’utente inserisce le cifre del numero binario un bit alla volta, partendo dal bit più significativo. Il programma dovrà visualizzare il numero decimale corrispondente. Suggerimento: per calcolare le potenze di 2 utilizzare la funzione pow includendo la libreria math.h.



## Implementazione

[Link al file in c](https://github.com/tommasosollo/INFORMATICA1/blob/main/ES02%20-%20Cicli%20e%20iterazioni/f_BinarioDecimale.c)

## Collaudo



Il programma funziona correttamente, infatti inserendo il numero binario 10101, la cui conversione in decimale è 2^0 \*1 + 2^1 \* 0 + 2^2 \* 1 + 2^3 \* 0 + 2^4 \* 1 = 1 + 4 + 16 = 21, il programma esegue correttamente la conversione in decimale.