

# Esercizi Blocco 0

Luca Oliveri  
luca.olivieri-1@unitn.it

Università di Trento — October 6, 2020

## Introduzione

Questa serie di file di esercizi verrà pubblicata mano a mano che progrediremo nel corso. Gli esercizi non saranno né valutati né corretti. Nonostante ciò, sono da considerarsi chiave per preparazione all'esame.



**Info:** Molti degli esercizi sono proposti dal sito `CodeStepByStep`. Previa registrazione, il sito propone un box dove si può direttamente scrivere il codice e poi eseguire un test automatico della sua correttezza. Il mio consiglio è, quando possibile, quello di sviluppare su VSCode sfruttando il sistema dei sotto-progetti, uno per esercizio. Copiare poi il codice su `CodeStepByStep` per verificarne la correttezza solo alla fine.

### 0.1 Somma a tre

Scrivere un programma completo che chiede all'utente tre numeri interi e ne calcola il prodotto.



**Occhio:** Se invece venisse chiesto di chiedere dieci numeri?

### 0.2

Scrivere un programma che chiede all'utente due numeri e ne calcola somma, prodotto, differenza, divisione e resto.

### 0.3

Scrivere un programma che scrive i numeri da 1 a 4 sulla stessa linea. Ripetere tre volte lo stesso output utilizzando i seguenti metodi.

- Usando una sola volta `cout` e separando gli spazi dai numeri Es: "3" e " ".
- Usando una sola volta `cout` mantenendo gli spazi insieme ai numeri Es: "3 ".
- Usando quattro `cout` separati.



**Estensione:** L'utente inserisce un numero e vengono stampati i numeri da 1 al numero inserito. Provate a capire quale tra le prime due opzioni è più efficiente.

### 0.4

Scrivere un programma che chiede all'utente di inserire due interi, e poi scrive il numero più grande seguito dalla string " is larger". Se i due numeri sono uguali, scrivi il messaggio " These numbers are equal".

### 0.5

Scrivere un programma che legge tre numeri e calcola la somma, la media, il prodotto, il più grande e il più piccolo.

## 0.6

Scrivere un programma che si fa inserire la lunghezza del raggio e calcola il diametro, circonferenza e area. Ci sono diversi modi nell'ottenere la costante  $\pi$ , tra cui scriversela a mano. Fare i calcoli direttamente nei comandi di `cout`.

## 0.7

Scrivere un programma che prende un intero e verifica se questo è pari o dispari e scrive qualcosa a proposito.

## 0.8

Scrivere un programma che prende due numeri interi e determina se il primo è un multiplo del secondo.

## 0.9

Scrivere un programma che stampa la tabella ASCII in questo formato, limitandosi alle lettere maiuscole, minuscole e le dieci cifre:

```
CHAR: A ASCII: 65
CHAR: B ASCII: 66
CHAR: C ASCII: 67
```



**Info:** Dato un intero `i`, il suo corrispondente carattere ASCII può essere ricavato tramite l'operatore di cast, quindi: `cout << (char) 65` stamperà A. Inoltre ricordare che invece scrivendo `int i = 'A'` il valore di `i` sarà 65.

## Esercizi CodeStepByStep

- `ifElseMystery1`
- `ifElseMystery2`
- `percentageGrade`
- `evenAverage`
- `fixErrors`
- `numberSquare`