

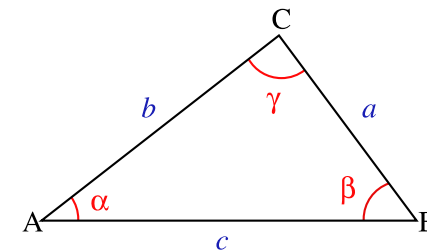


# Esercitazione 10 (12-03 \*)



## 10.1 Classe dei triangoli in C++

- Scrivere una classe `Triangle` per rappresentare dei triangoli
- Parametri del costruttore: lunghezze dei lati `a`, `b`, `c`
- Fornire metodi `perimeter` e `area` (Erone)
- Nella parte principale del programma:
  - Creare un oggetto triangolo, con lati forniti all'utente
  - Invocare i metodi `area` e `perimeter` sull'oggetto
  - Infine mostrare l'area ed il perimetro del triangolo



*Formula di Erone:*  $area = \sqrt{s * (s-a) * (s-b) * (s-c)}$   
*Con*  $s = (a + b + c) / 2$ , *semiperimetro*

[http://en.wikipedia.org/wiki/Heron%27s\\_formula](http://en.wikipedia.org/wiki/Heron%27s_formula)

*Per* `sqrt`, *includere* `<cmath>`



## 10.2 Cifrario di Cesare in C++

- Ogni lettera è sostituita dalla lettera che la segue nell'alfabeto dopo 3 posti (o n posti, *chiave* del cifrario)
  - A sostituita da D; B da E ecc.
  - Per le ultime lettere, si ricomincia a ruota, dalla A
  - Chiaro ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - Cifrato DEFGHIJKLMNOPQRSTUVWXYZABC
- Chiedere all'utente n, *chiave* del cifrario
- Cifrare l'input da console, riga per riga, e scriverlo in un file
- Cifrare le sole lettere minuscole e maiuscole
  - Lasciare inalterati gli altri caratteri



*Ricordarsi che in C++ vale: `char('a' + 3) == 'd'`*



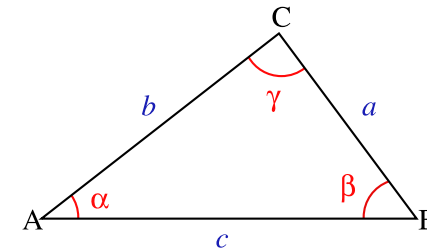
## 10.3 Conteggio delle lettere in C++

- Leggere un intero testo da un file
- Calcolare la percentuale di ogni singola lettera dell'alfabeto
  - Quante **A**? Quante **B**? ecc.
- Considerare nel totale le sole lettere dell'alfabeto, senza cifre e altri simboli
- Conteggiare assieme le stesse lettere, scritte sia in maiuscolo che minuscolo
  - **A** assieme ad **a**; **B** assieme a **b**; ecc.



## 10.4 Gestione triangoli in C++

- Riusare la classe `Triangle` dell'esercizio 10.1
- Gestire una lista di triangoli, come `vector<Triangle*>`
- Ciclicamente, permettere all'utente di effettuare una delle seguenti operazioni:
  - Aggiungere un nuovo triangolo
  - Rimuovere un triangolo esistente, dato il suo indice (ricordarsi `delete`)
  - Calcolare la somma delle aree di tutti i triangoli



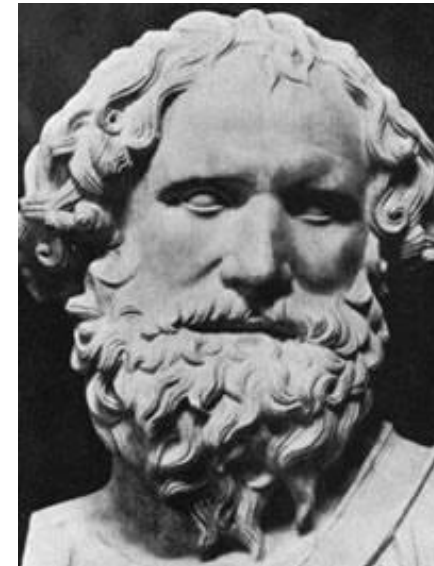
```
delete triangles[i]; // remove triangle from memory  
triangles.erase(begin(triangles) + i); // remove pointer from vector
```

C++



## 10.5 Crivello di Eratostene in C++

- Definire e testare funzione che calcola la lista di numeri primi fino ad **n**
  - Parametro: **n**
  - Risultato: `vector<int>`
- Algoritmo
  - Inserire i numeri da 2 ad **n** in una lista
  - Per ciascun numero **x** nella lista...
  - Togliere dalla lista i multipli di **x** (escluso **x** stesso)
  - I numeri rimanenti sono i numeri primi cercati



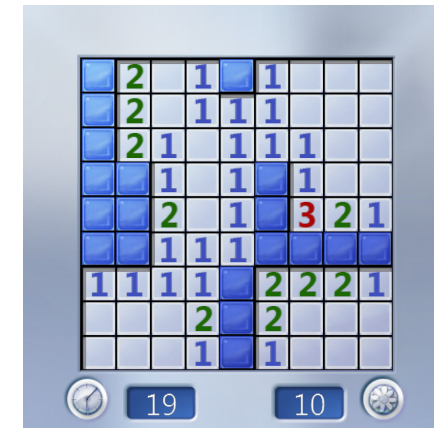
*Usare il metodo `erase` di `vector`, oppure usare un `vector<bool>`  
Attenzione, non rimuovere o inserire elementi in un ciclo `for (val : seq)`*

*Esempio Python disponibile su GitHub*



## 10.6 Campo minato, bandiere

- Consentire all'utente di posizionare bandiere su celle *coperte*
  - L'utente suppone la presenza di una mina
- Implementare il metodo `flag_at` di `BoardGame`



## 10.7 Campo minato, automatismi

- Se l'utente usa il metodo `flag_at` su una cella *già scoperta*
  - Se attorno ci sono bandiere pari alle mine contate
  - ⇒ Chiamare `play_at` sulle celle coperte attorno
- Se l'utente usa il metodo `flag_at` su una cella *già scoperta*
  - Se attorno ci sono bandiere e celle coperte pari alle mine contate
  - ⇒ Chiamare `flag_at` sulle celle coperte attorno

