

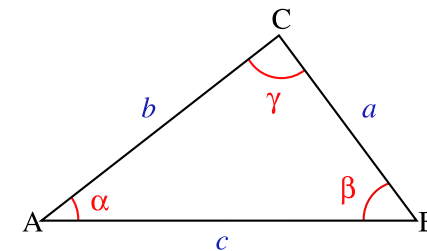


Esercitazione 2 (10-08)



2.1 Funzione, Erone

- Definire una funzione **heron** per il calcolo dell'area di un triangolo
 - Parametri: tre lati come **float**
 - Risultato: area come **float**
- Invocare la funzione dalla shell interattiva
- Aggiungere poi al programma una funzione **main**
 - *Procedura, senza parametri e senza risultato*
 - Chiedere all'utente tre valori (chiamando **input**)
 - Poi chiamare **heron** con questi parametri
 - Infine mostrare all'utente il risultato (chiamando **print**)



Formula di Erone: $area = \sqrt{s * (s - a) * (s - b) * (s - c)}$

Con $s = (a + b + c) / 2$, semiperimetro

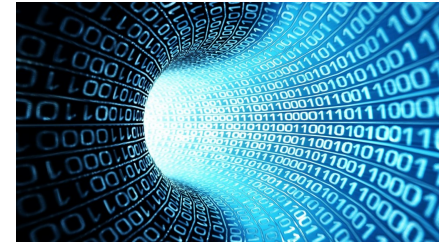
http://en.wikipedia.org/wiki/Heron%27s_formula



2.2 Conteggio di 0 e 1

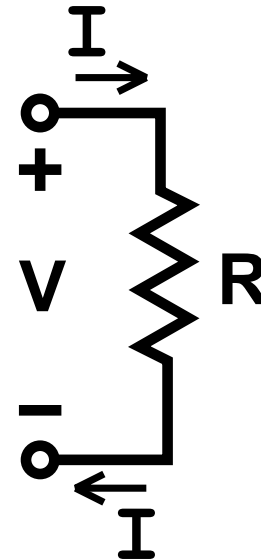
- Definire una funzione che conta separatamente il numero di cifre **0** ed **1** presenti in un testo
 - Parametro: stringa di testo da elaborare
 - Risultato: tupla dei due valori conteggiati
- Invocarla da una funzione **main**
 - I/O con l'utente solo nella funzione **main**

*Nella funzione, usare un ciclo **for** sulla stringa*



2.3 Classe dei cavi

- Definire una classe per rappresentare cavi di rame
- Costruttore con parametri: lunghezza (in *m*) e sezione (in *mm²*)
- Campo costante per resistività
 - $\rho = 0.0171 \text{ } \Omega \cdot \text{mm}^2 / \text{m}$
- Metodo per calcolare la resistenza (in *Ohm*)
 - $\rho * \text{length} / \text{area}$
- Metodo per calcolare l'intensità di corrente (in *Ampere*), data la tensione (parametro, in *Volt*)
 - $\text{voltage} / \text{resistance}$



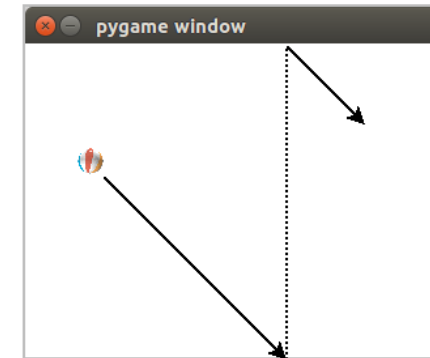
2.4 Triangolo di cerchi colorati

- Chiedere all'utente un numero **n**
- Generare **n** righe di cerchi, allineate a sinistra
- Il numero di riga corrisponde al numero di cerchi da disegnare
- Ogni cerchio ha un colore casuale



2.5 Animazione pallina

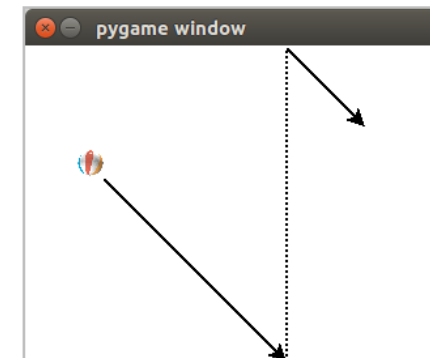
- Partire dall'esempio dell'animazione orizzontale, senza classi
 - Aggiungere ad ogni turno anche uno spostamento verticale
 - Quando la pallina arriva ad uno dei 4 bordi...
 - Ricompare dal bordo opposto
- Direzione di movimento scelta dall'utente
 - All'inizio chiedere all'utente **dx** e **dy**
 - Ciascuno limitato tra -5 e +5
- Mostrare l'animazione di una pallina su un canvas



2.6 Conteggio fotogrammi

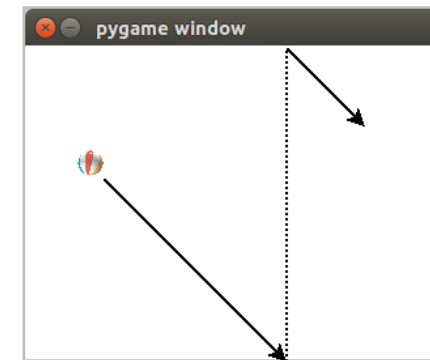
- Ripetere l'esercizio 2.5, senza classi, ma fermare la pallina dopo 5 fotogrammi
- Alla pressione di un tasto, se la pallina è ferma:
 - Far ripartire l'animazione, per soli altri 5 fotogrammi

Usare una variabile globale per contare i fotogrammi



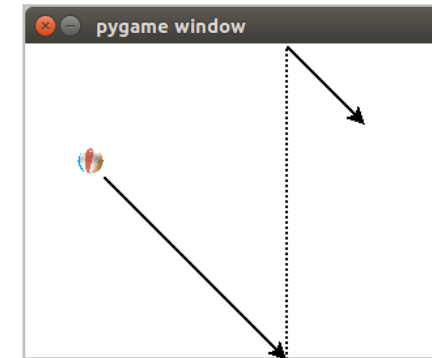
2.7 Pallina modificata

- Partire dalla classe `Ball` vista a lezione
- Modificare il metodo `move`
 - Quando la pallina arriva ad uno dei 4 bordi...
 - Ricompare dal bordo opposto



2.8 Pallina con conteggio

- Definire una propria classe `Ball`
- Incapsula il movimento dell'esercizio 2.6
 - Campi `x`, `y`, `dx`, `dy`
 - Metodi `move` e `position`
 - Campo per contare i fotogrammi
 - Metodo `go` per far ripartire la pallina (solo se ferma)
- Mostrare l'animazione di una pallina su un canvas
 - Alla pressione di un tasto, viene chiamato il metodo `go`



Il metodo `go` riavvia solo il conteggio, ma non sposta la pallina



2.9 Albero di cerchi colorati

- Chiedere all'utente un numero n
- Generare n righe di cerchi, allineate al centro
- Il numero di riga corrisponde al numero di cerchi da disegnare
- Ogni cerchio ha un colore casuale
- Aggiungere in fondo una riga con un solo cerchio

