

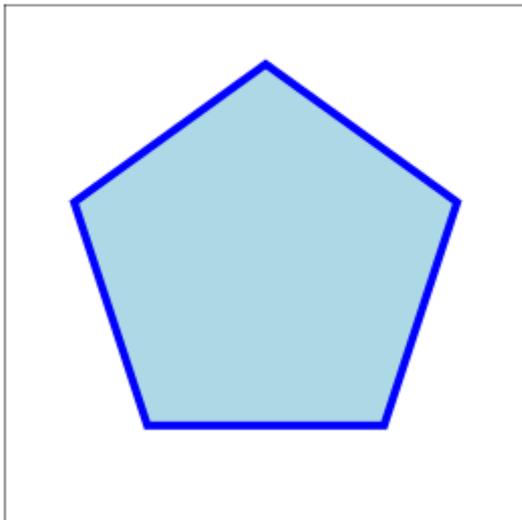
ESERCIZIO 1 – SPERIMENTARE I CANVAS HTML5 (E RIPASSARE JAVASCRIPT)

Utilizzando i Canvas HTML5 e JavaScript creare un'applicazione interattiva che disegni un poligono regolare con un numero di lati a scelta, compreso fra 3 e 12.

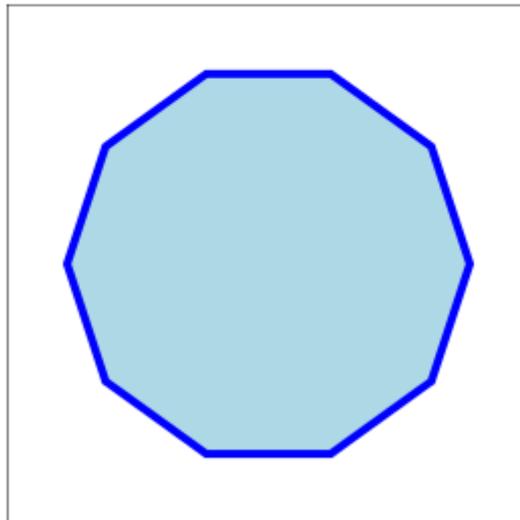
L'applicazione deve permettere di scegliere il numero di lati da un menu a tendina e visualizzare il poligono corrispondente.

Esempio:

Numero di lati: ▼



Numero di lati: ▼



Suggerimento: per disegnare i poligoni occorre usare le istruzioni moveTo (per il primo vertice) e.lineTo (dal secondo in poi), calcolando le coordinate dei vertici x, y a partire da:

- il centro del poligono (il punto centrale del canvas)
- il raggio del poligono (a scelta, purché il poligono stia all'interno canvas)
- l'angolo rispetto al centro del poligono (in radianti); conviene posizionare il primo vertice a $-\pi/2$, gli altri vertici si ottengono dividendo la circonferenza per il numero di lati

$$X = \text{centerX} + \text{raggio} * \cos(\text{angolo in radianti})$$

$$Y = \text{centerY} + \text{raggio} * \sin(\text{angolo in radianti})$$

ESERCIZIO 2 – SPERIMENTARE LA MANIPOLAZIONE DEI PIXEL

Utilizzando i Canvas HTML5 e JavaScript, creare un'applicazione interattiva che applichi dei filtri a scelta ad un'immagine (vedere l'esempio).

L'applicazione deve permettere di scegliere il tipo di filtro e visualizzare l'immagine corrispondente.

Esempio:

Click on the buttons below to filter the image.



Suggerimenti:

- Invert: invertire le tre componenti di colore
- Gray: media pesata delle tre componenti di colore, in cui il verde e il rosso pesano più del verde ($\text{Gray} = 0.2126 * R + 0.7152 * G + 0.0722 * B$)
- Brighten: aggiungere un valore costante a tutte le componenti di colore
- Threshold: convertire in grey e poi applicare una soglia (al di sopra il valore è 255, al di sotto 0)
- Blue: azzerare le componenti rossa e verde
- Green: azzerare le componenti rossa e blu

- Red: azzerare le componenti blu e verde
- Yellow: azzerare la componente blu e invertire le altre due
- Cyan: azzerare la componente rossa e invertire le altre due
- Magenta: azzerare la componente verde e invertire le altre due
- Reset: ricaricare l'immagine originale

Facoltativo: aggiungere filtri più complessi

I filtri precedenti possono essere calcolati pixel per pixel. Nel caso di filtri più complessi (e.g. blur, sharp, edge detect, ...) il nuovo valore di ogni pixel dipende dai pixel circostanti, ed è quindi necessario applicare una matrice di convoluzione.

Per saperne di più, guardare ad esempio: <http://lodev.org/cgtutor/filtering.html>