

# **Python #2 Esercizi con blocchi condizionali**

esercizi utili per testare le proprie conoscenze nell'ambito dei linguaggi di programmazione strutturata relativamente all'utilizzo dei blocchi condizionali.

---

## **Python #1 Esercizi Sequenziali**

esercizi utili per testare le proprie conoscenze nell'ambito dei linguaggi di programmazione strutturata relativamente all'utilizzo dei blocchi condizionali.

---

## **Scilab #3 Elaborazione di una immagine in Scilab: aumentare o diminuire la luminosità di una foto**

Effettuare una semplice elaborazione di una immagine in scilab con l'obiettivo di aumentare o diminuire la luminosità di una foto.

---

# **Scilab #3 Metodo dei rettangoli (Riemann) per determinare il valore di un integrale definito**

Determinare il valore di un integrale definito utilizzando il metodo dei rettangoli e sfruttando il software di calcolo numerico Scilab.

---

# **Scilab #2 Metodo di Bisezione per determinare gli zeri di una funzione**

Determinare gli zeri di una funzione utilizzando il metodo di bisezione e sfruttando il software di calcolo numerico Scilab.

---

# **Scilab #1 Analisi di Funzioni**

# Matematiche Elementari

Analizzare alcune semplici funzioni elementari utilizzando il software di calcolo numerico Scilab.

---

## Programmare Arduino – Le basi

Arduino è collegabile al PC tramite un cavo USB. L'IDE (Integrated Development Environment) è l'ambiente in cui si scrive e si carica il codice sulla scheda Arduino (scaricabile dal sito ufficiale di Arduino (<https://www.arduino.cc/>)). E' possibile anche usare l'*Arduino Web Editor* per programmare online, anch'esso disponibile nel sito ufficiale di Arduino.

Un programma Arduino è chiamato sketch e usa un linguaggio simile a C/C++. Ogni sketch ha due funzioni fondamentali (spiegate nel dettaglio qui)

- **setup()**: Configura i pin o inizializza componenti (eseguito una sola volta all'inizio).
- **loop()**: Contiene il codice che viene eseguito continuamente.

[crayon-6ab164bfd5c7b847788291/]

Ogni funzione e in generale ogni blocco di codice, viene racchiuso tra parentesi graffe { e }. Il totale delle parentesi aperte { deve essere lo stesso di quelle chiuse }.

Ogni istruzione di programma termina con un punto e virgola ;

E' possibile inserire dei commenti nel codice, utilizzando la doppia barra // per commentare su una linea e barra asterisco (e asterisco barra) /\* ... \*/ per commenti su più

linee.

Nella funzione **setup()**, vengono tipicamente configurati i pin. La funzione da usare è **pinMode**. Ad esempio:

```
[crayon-6ab164bfd5c85733684256/]
```

```
[crayon-6ab164bfd5c87565423498/]
```

Altre funzioni importanti e ricorrenti sono:

- **digitalWrite(pin, valore)**: Imposta un pin su HIGH o LOW.
- **digitalRead(pin)**: Legge lo stato di un pin digitale (HIGH o LOW).
- **analogWrite(pin, valore)**: Imposta un'uscita PWM (valori tra 0 e 255).
- **analogRead(pin)**: Legge un valore analogico da un pin (da 0 a 1023).
- **delay(millisecondi)**: Imposta una pausa nell'esecuzione del programma. La pausa è espressa in millisecondi

Esistono molte altre funzioni, che vengono illustrate con gli esempi presenti nel sito.

Per altre funzioni, si rimanda ai tutorial ufficiali di Arduino.

---

## JAVASCRIPT #6 Realizzazione di una Avventura Testuale, CSS e JAVASCRIPT

Realizzare un programma che implementi una semplice avventura testuale utilizzando il linguaggio di programmazione HTML, CSS e JAVASCRIPT.

---

# **JAVASCRIPT #5 Realizzazione del gioco della Roulette HTML, CSS e JAVASCRIPT**

Realizzare un programma che implementi il gioco della Roulotte utilizzando il linguaggio di programmazione HTML, CSS e JAVASCRIPT.

---

# **JAVASCRIPT #4 Realizzazione del gioco del TRIS HTML, CSS e JAVASCRIPT**

Realizzare un programma che implementi il gioco del TRIS utilizzando il linguaggio di programmazione HTML, CSS e JAVASCRIPT.