

**ITIS-LS “Francesco Giordani” Caserta**

**prof. Ennio Ranucci**

**a.s. 2019-2020**

***Analizziamo gli obiettivi e troviamo soluzioni da codificare in Python***

Esercizi svolti in ambiente Python



Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018)



**ITIS-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es0**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:**

La struttura che controlla il flusso delle istruzioni: Selezione (if else)

**Obiettivo del programma:**

Controllare se è maggiorenne o minorenni.

```
età=18
```

```
if età >= 18:
```

```
    print("sei maggiorenne")
```

```
else:
```

```
    print("sei minorenni")
```

**ITIS-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es1**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:**

Le strutture che controllano il flusso delle istruzioni: Selezione (if else) e Iterazione indefinita(While)

**Obiettivo del programma:**

Scrivere un programma che, leggendo due numeri interi, sottrae il minore dal maggiore finché la loro differenza diventa inferiore a 3 unità, visualizzando sullo schermo il risultato di ogni iterazione.

```
a=int(input('Inserisci il numero a: '))
```

```
b=int(input('Inserisci il numero b: '))
```

```
if a<b:
```

```
    a,b=b,a
```

```
    d=a-b
```

```
while d>=3:
```

```
    print(d)
```

```
    d=d-b;
```

**ITIS-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es2**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:**

La struttura che controlla il flusso delle istruzioni: Iterazione definita (for)

**Obiettivo del programma:**

Stampare a video i numeri interi da 1 a 10

```
for count in range(1,11):
```

```
    print (count)
```

**ITIS-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es3**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:**

La struttura che controlla il flusso delle istruzioni: Iterazione definita (for); Lista

**Obiettivo del programma:**

Sommare gli elementi presenti in una lista tramite il ciclo for

```
# definizione della lista
```

```
fibonacci = [1,1,2,3,5,8,13,21,34,55,89,144]
```

```
# variabile per l'archiviazione della somma
```

```
totale = 0
```

```
# sommatoria dei valori in lista
```

```
for val in fibonacci:
```

```
    totale = totale + val
```

```
# stampa della somma ottenuta
```

```
print(totale)
```

**ITIS-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es4**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:** Struttura che controlla il flusso delle istruzioni: Iterazione definita (for); Lista

**Obiettivo del programma:** Sommare gli elementi presenti in una lista tramite il ciclo for escludendo tutti i valori inferiori a 3

```
# definizione della lista
```

```
fibonacci = [1,1,2,3,5,8,13,21,34,55,89,144]
```

```
# variabile per l'archiviazione della somma
```

```
totale = 0
```

```
# sommatoria dei valori in lista
```

```
for val in fibonacci:
```

```
    # controllo sul valore degli elementi sommati
```

```
    if val > 3:
```

```
        totale = totale + val
```

```
# controllo sul totale ottenuto
```

```
if totale == 369:
```

```
    # stampa della somma ottenuta
```

```
    print("Il totale della somma è " + str(totale))
```

```
else:
```

```
    # istruzione alternativa in caso
```

```
    # di esito negativo del controllo
```

```
    print ("Valore differente da quello atteso.")
```

**ITIS-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es5**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:** Struttura che controlla il flusso delle istruzioni: Iterazione indefinita (do while)

**Obiettivo del programma:** Chiedi all'utente di inserire numeri finchè indovina

while True:

```
    numeroFortunato = int(input('Inserisci un numero da 1 a 10: '))
```

```
    if numeroFortunato == 8:
```

```
        print('Hai indovinato!')
```

```
        break # numero indovinato, interrompi il ciclo
```

```
    else:
```

```
        print('Ritenta sarai più fortunato')
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico: 2019/2020**

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio: es6**

**Versione: 1.0**

**Programmatore/i:**

**Sistema Operativo: Windows 10**

**Compilatore/Interprete: Python 3.7.0**

**Obiettivo didattico:** Struttura che controlla il flusso delle istruzioni: Iterazione indefinita (do while)

**Obiettivo del programma:** Chiedi all'utente di inserire numeri finchè indovina

```
i = 1
```

```
while True:
```

```
    print(i)
```

```
    i = i + 1
```

```
    if(i > 3):
```

```
        break
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es7

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Struttura che controlla il flusso delle istruzioni: Iterazione indefinita (while)

**Obiettivo del programma:** Stampa i numeri pari tra 10 e 20

```
val = 10
```

```
while val < 20:
```

```
    print (val)
```

```
    val+=2
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es8

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Divide et impera(top down) (funzioni)

**Obiettivo del programma:** Scrivere le funzioni Ciao. Area, saluta. "ciao" è una funzione senza parametri che scrive "Ciao ragazzi" a video, "area" è una funzione che riceve in ingresso la base e l'altezza e restituisce l'area. "saluta" è una funzione che riceve in ingresso un nome e restituisce la stringa "Buongiorno" concatenata al valore del nome.

```
def ciao():
```

```
    print ("Ciao ragazzi")
```

```
def area(base,altezza):
```

```
    return base*altezza
```

```
def saluta(nome):
```

```
    print ("Buongiorno",nome)
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es9

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Divide et impera(top down) (funzioni)

**Obiettivo del programma:** Definire una funzione che calcola il fattoriale.

```
def fatt(n):  
    if n <= 1:  
        return 1  
    return n*fatt(n-1)  
print ("2! = ",fatt(2))  
print ("3! = ",fatt(3))  
print ("4! = ",fatt(4))  
print ("5! = ",fatt(5))
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es10

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** I Grafici – La modalità grafica

**Obiettivo del programma:** Disegnare i punti (1,1) (2,4) (3,9) (4,16)

*Note:*

*installazione della libreria grafica:*

*cmd*

*portarsi nella dir cd C:\Users\ennio\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Python  
3.7*

*python -m pip install -U pip*

*python -m pip install -U matplotlib*

Note:

**plot**

*plot(y)*, un grafico con ascisse automatiche (da 0 a len(y)-1) e ordinate y

*plot(x, y)*, un grafico con ascisse x e ordinate y

*plot(x, y, "ro")*, attributi in forma abbreviata (r=red, o=circle)

**axis** definisce la dimensione del riquadro grafico

```
import matplotlib.pyplot as plt
```

```
plt.plot([1,2,3,4], [1,4,9,16], 'ro')
```

```
plt.axis([0, 6, 0, 20])
```

```
plt.show()
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es11

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** I Grafici – La modalità grafica

**Obiettivo del programma:** Generare 50 numeri equidistanti tra -1 e 1 Disegnare la funzione  $y=x^{**3}$

Note:

*NumPy è una libreria open source per il linguaggio di programmazione Python, che aggiunge supporto a grandi matrici e array multidimensionali insieme a una vasta collezione di funzioni matematiche di alto livello per poter operare efficientemente su queste strutture dati.*

*Ci sono un numero enorme di funzioni predefinite in numpy che calcolano automaticamente diverse quantità. Ad esempio: mean(): calcola la media di un vettore o matrice; sum(): calcola la somma di un vettore o matrice; std(): calcola la deviazione standard; min(): trova il minimo nel vettore o matrice; max(): trova il massimo;*

*numpy.linspace(start, start, num, 50, endpoint): Restituisce numeri equidistanti in un intervallo specificato[start, stop]. Il punto finale dell'intervallo può essere facoltativamente escluso.*

**PARAMETRI**

**start:** Valore iniziale della sequenza;

**stop:** Valore finale della sequenza, a meno che l'endpoint non sia impostato su False.

**num :** int, opzionale. Numero di campioni (numeri) da generare. Il valore predefinito è 50. Deve essere non negativo.



**endpoint** : bool, opzionale. Se True, stop è l'ultimo campione. In caso contrario, non è incluso. Il valore predefinito è True.

```
import numpy as np

import matplotlib.pyplot as plt

x = np.linspace(-1.0,1.0,50,endpoint=True)

y = x**3

plt.plot(x,y)

plt.show()
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es12

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Tecnica della somma

**Obiettivo del programma:** Sommare gli elementi numerici di una lista  
def somma(lista):

```
    risultato = 0
```

```
    for numero in lista:
```

```
        risultato += numero
```

```
    print('Il risultato della somma è... ' + str(risultato))
```

```
somma([1,2,3])
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es13

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Le liste

**Obiettivo del programma:** Dato un carattere riconoscere le vocali

```
def cerca_vocali(carattere):
```

```
    vocali = "aeiou"
```

```
    if carattere in vocali:
```

```
        print('Il carattere ' + carattere + ' è una vocale')
```

```
    else:
```

```
        print('Il carattere ' + carattere + ' non è una vocale')
```

```
cerca_vocali('d')
```

```
cerca_vocali('e')
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es14

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Le liste

**Obiettivo del programma:** Data una lista di numeri rappresentare il relativo istogramma

```
def istogramma(lista):
```

```
    for numero in lista:
```

```
        print("*" * numero)
```

```
lista = [3,7,9,5]
```

```
istogramma(lista)
```

ITI-LS "Francesco Giordani" Caserta

Anno scolastico: 2019/2020

Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni

Data:

Numero progressivo dell'esercizio: es15

Versione: 1.0

Programmatore/i:

Sistema Operativo: Windows 10

Compilatore/Interprete: Python 3.7.0

Obiettivo didattico: Le liste

**Obiettivo del programma:** In Svezia, i bambini giocano spesso utilizzando un linguaggio un pó particolare, detto "rövarspråket", che significa "linguaggio dei furfanti": consiste nel raddoppiare ogni consonante di una parola e inserire una "o" nel mezzo. Ad esempio la parola "mangiare" diventa "momanongogarore". Scrivi una funzione in grado di tradurre una parola o frase passata tramite input() in "rövarspråket".

```
def traduttore():
```

```
    print("""
```

```
    Ciao! raddoppio ogni consonante delle parole e metto una "o" in mezzo...
```

```
    """)
```

```
    vocali = "aeiou"
```

```
    specials = [" ", ",", ".", "?", "!", "'", "",""]
```

```
    while True:
```

```
        testo = input("\nInserisci il testo che desideri tradurre -> ')
```

```
        tradotta = ""
```

```
        for x in testo:
```

```
            if x in vocali or x in specials:
```

```
                tradotta += x
```

```
            else:
```

```
                tradotta = tradotta + x + "o" + x
```

```
        print(f"Traduzione! '{tradotta}'")
```

```
        if input("\nDesidere tradurre un'altra frase? (no per uscire) ") == "no":
```

```
            break
```

```
traduttore()
```

ITI-LS "Francesco Giordani" Caserta

Anno scolastico: 2019/2020

Classe 3<sup>a</sup> sez.B spec. Informatica e telecomunicazioni

Data:

Numero progressivo dell'esercizio: es16

Versione: 1.0

Programmatore/i:

Sistema Operativo: Windows 10

Compilatore/Interprete: Python 3.7.0

Obiettivo didattico: I vettori e le matrici, la libreria numpy e il calcolo vettoriale e matriciale.

Obiettivo del programma: Operazioni con le matrici

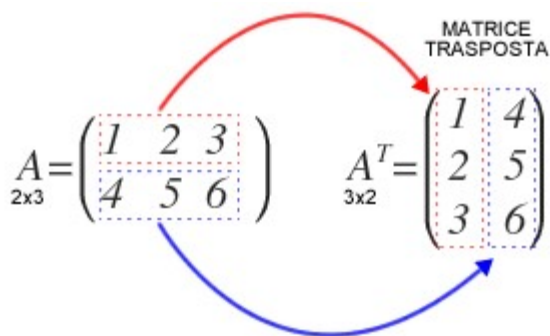
Prodotto di matrici

La prima matrice deve avere il numero di colonne uguali al numero di righe della seconda matrice.

Altrimenti il prodotto riga per colonna è impossibile. Per moltiplicare le due matrici uso la funzione dot().

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \cdot \begin{pmatrix} 5 & 6 & 7 \\ 8 & 9 & 10 \end{pmatrix} = \begin{pmatrix} 21 & 24 & 27 \\ 47 & 54 & 61 \end{pmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad \text{e} \quad \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \\ = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$



WWW.ANDREAMININI.ORG

```
import numpy as np
```

```
vetNum=[1, 2, 3]
```

```
print ("Stampa vettore: ",vetNum)
```

```
print("il primo elemento del vettore: ", vetNum[0])  
  
matNum=[[1,2,3],[4,5,6]]  
  
print("Stampa matrice: ",matNum)  
  
print(" elemento 1 1 della matrice: ",matNum[1][1])  
  
print(" elemento 0 2 della matrice: ",matNum[0][2])  
  
#Questo tipo di dato mi consente di usare la variabile m  
  
#per compiere le operazioni del calcolo vettoriale e matriciale.
```

```
print ("stampa vettore Ones np ")  
  
vetOnes=np.ones(5)  
  
print(vetOnes)  
  
print ("stampa vettore Ones np moltiplicato per due ")  
  
print(2*vetOnes)  
  
print ("stampa vettore Ones sommato a se stesso 3 volte ")  
  
print(vetOnes+vetOnes+vetOnes)  
  
print ("stampa matrice np ")  
  
mat = np.array([[1,2],[3,4]])  
  
print(mat)  
  
print ("stampa della matrice np sommata a se stessa ")  
  
print(mat+mat)  
  
print ("Prodotto di due matrici ")  
  
a=np.array([[1,2],[3,4]])  
  
print(a)  
  
b=np.array([[5,6,7],[8,9,10]])  
  
print(b)  
  
axb=np.dot(a,b) #prodotto di due matrici  
  
print("matrice prodotto : ")  
  
print (axb)
```

```
print ("Matrice e matrice trasposta ")
m=np.array([[1,2,3],[4,5,6]])
print (m)
print(np.transpose(m))
```

**ITI-LS "Francesco Giordani" Caserta**

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es17

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** Il riconoscimento delle parti di un testo.

**Obiettivo del programma:** Dividere il testo in token

```
import shlex
text = """Riconosci le parti di questo testo."""
print ('ORIGINAL:', repr(text))
print
print ('TOKENS:')
print (shlex.split(text))
```

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es18

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** La grafica della tartaruga

**Obiettivo del programma:** Cambiare il colore di sfondo

```
from turtle import Turtle, Screen
tartaruga = Turtle()
sfondo = Screen()
sfondo.colormode(255)
```

R = 255  
G = 255  
B = 0  
sfondo.bgcolor((R, G, B))

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es19

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** La grafica della tartaruga

**Obiettivo del programma:** Disegnare un cerchio in loop infinito

```
from turtle import Turtle, Screen
turtle = Turtle()
while True:
    turtle.forward(1)
    turtle.right(1)
```

**Anno scolastico:** 2019/2020

**Classe 3<sup>^</sup> sez.B spec. Informatica e telecomunicazioni**

**Data:**

**Numero progressivo dell'esercizio:** es20

**Versione:** 1.0

**Programmatore/i:**

**Sistema Operativo:** Windows 10

**Compilatore/Interprete:** Python 3.7.0

**Obiettivo didattico:** La grafica della tartaruga

**Obiettivo del programma:** Scrivere i numeri fino a 7

```
from turtle import Turtle, Screen
tartaruga = Turtle()
sfondo = Screen()
sfondo.colormode(255)
tartaruga.penup()
for i in range(8):
    tartaruga.write(i)
    tartaruga.forward(20)
```