Incontro II

Giovedì 3 ottobre

9 Uso di while

```
x = 0
while x <= 10 :
    x += 1
print(x)
```

Ripete “aggiungi uno a x” fino a che x non è minore o uguale a 10. Quindi l’output è 11. Un caso un po’ più articolato:

```
vero = True
delta = 0.02
conta = 0
while vero :
    x = random.random()
    y = random.random()
    if abs(x-y) <= delta :
        vero = False
    conta += 1
conta
```

Problema 7. Disegna per punti il grafico della parabola di equazione $y = ax^2 + bx + c$. Imposta a, b, c.

Problema 8. Disegna il grafico della parabola di equazione $y = ax^2 + bx + c$. Imposta a, b, c. Nel grafico il vertice della parabola deve trovarsi al centro rispetto a x

Problema 9. Considera il quadrato $ABCD$ di lato $AB = 1$ inscrivi in esso un arco parabolico $\mathcal{P}_{AB}$ di estremi A e B e vertice nel punto medio di CD.

Conviene utilizzare la parabola $\mathcal{P}$ di equazione $y = 1 - 4x^2$, inscritta nel quadrato di lato AB, dove $A = \left(-\frac{1}{2}, 0\right)$ e $B = \left(\frac{1}{2}, 0\right)$.

Stima l’area del segmento parabolico.

Problema 10. Se si lancia una moneta 2 volte, la probabilità di ottenere una testa e una croce (in qualsiasi ordine) è pari al 50 %. Se la moneta viene lanciata 4 volte, la probabilità di ottenere due teste e due croci, in qualsiasi ordine, è ancora pari al 50 %? Costruisci un programma che ti permetta di rispondere alla domanda.

Problema 11. Zig-modo è piatto (2 dimensioni) per noi che lo vediamo da “fuori” per i zig-abitanti è il mondo.

Il generico zig-abitante si muove in modo un po’ strano (per noi, naturale per lui): prima di fare un passo lancia il suo zig-dado (un quadrato con i lati colorati diciamo: colore 1, colore 2, colore 3, colore 4, se

- esce 1 avanti uno;
- esce 2 indietro uno;
- esce 3 ruota di 90^o e avanti uno;
- esce 4 ruota di -90^o e avanti uno

Costruisci una zig-passeggiata

Problema 12. Dato un triangolo equilatero ABC sia P_0 un fissato punto interno, si scelga a **caso** uno dei tre vertici del triangolo (ad esempio A), sia P_1 il punto medio del segmento P_0A. Il punto P_1, ora, assume il ruolo di P_0 nella costruzione precedente, perciò: si sceglie a caso un vertice del triangolo e si costruisce P_2.

Ripetere la costruzione per n volte disegnando i punti così ottenuti.

10 Comandi ... per oggi I

10.1 Liste

Le liste in Python sono collezione di elementi:

```
cose = ['dado','bullone','fascetta','vite','ranella']
numeri = [12,15,59,93,17,1]
misto = ['gatto',23,12.5,'a','zeta']
```

10.2 Operazioni con le liste

```
# aggiungere in coda
cose.append('brugola')
```

```
# ordinare
numeri.sort()
```

11 Esercizi

Problema 13. Utilizza le operazioni, *append* e *sort* per tutte le liste create.

Problema 14. Genera casualmente una lista di 100 caratteri e poi ordinala.
Stampa la lista non ordinata e quella ordinata.

```
# caratteri ascii A-Z da 65 a 90
# caratteri ascii a-z da 97 a 122
B = chr(66)
g = chr(103)
```

12 Comandi ... per oggi II

12.1 Vettori array

```
import numpy as np
```

```
numeri = np.array(numeri, dtype = int)
```

Numpy

12.2 Operazioni con i vettori

```
numeri = np.array(numeri, dtype = int)
inumeri = np.zeros(6, dtype = int)
```

```
for k in range(6) :
   x = random.randint(15,139)
   inumeri[k] = x
```

```
n = numeri + inumeri
```

```
n = numeri * inumeri
```

```
n = np.dot(numeri,inumeri)
```

12.3 Qualche idea in più su array

```
import numpy as np
import random

Vint = np.zeros(10, dtype = int)    # array di interi
Vfloat = np.zeros(10, dtype = float) # array di float
##
print(Vint, Vfloat)
##
prove = 20
for k in range(prove) :
    j = random.randint(0,9)
    i = random.randint(0,9)
    x = random.randint(35,55)
    y = random.random()
    Vint[j] += x
    Vfloat[i] += y
##
print(Vint, Vfloat)
```

13 Esercizio

Problema 15. Genera 10 vettori tridimensionali (x, y, z) e ricava il prodotto scalare di tutte le coppie possibili di vettori $\binom{10}{2}$ [2].

14 Comandi . . . per oggi III

14.1 Grafica in Sympy e uso di funzioni

```
from sympy import *
init_printing()
```

```
#parabola
def p(x) :
    return x**2-4*x
#retta
def r(x) :
    return 2*x-6
```

```
x = symbols('x')
Gp = plot(p(x), (x,-1,6), show = False)
Gr = plot(r(x), (x,-1,6), show = False)
Gp.append(Gr[0])
Gp.show()
```

14.2 Grafica con vettori

```
 n = 1000
figura = plt.figure(figsize=(6,3))
plt.axis('equal')
punti = []
for k in range(n) :
    x = 4*random.random()-2
```

[2] Usando sympy si ricava 45 con il comando: binomial(10,2)

```
    y = 2*random.random()-1
    r = random.random()
    g = random.random()
    b = random.random()
    punti.append([x,y,r,g,b])
tot = len(punti)
punti = np.array(punti,dtype=np.float64)
iColori = np.zeros((tot,4))
iColori[:,0] = punti[:,2]
iColori[:,1] = punti[:,3]
iColori[:,2] = punti[:,4]
iColori[:,3] = 1.0
plt.scatter(punti[:,0], punti[:,1], 0.8, color=iColori)
plt.show()
```

15 Esercizi

Problema 16. La porta degli zombie è una fessura di larghezza ℓ, in un muro impenetrabile, incessantemente migliaia di zombie camminano in modo disordinato verso questo muro e a ogni secondo uno di essi lo raggiunge in un punto aleatorio.

Solo gli zombie fortunati si trovano di fronte alla porta e possono superarla, gli altri invece urtano contro il muro e si distruggono.

Sia ℓ la larghezza della porta e d la larghezza del muro, t_{oss} il tempo, in secondi, di osservazione.

Costruisci una statistica, in cui registrare la frequenza di zombie che attraversano la porta nel I, II, III . . . , C (centesimo) secondo.

È necessario ripetere molte volte l'osservazione per stimare quanto richiesto.

Output sia numerico che grafico.

Problema 17. Due persone A e B partono contemporaneamente dagli estremi di una strada rettilinea, dirette una incontro all'altra. La distanza iniziale è $d = 15\ km$; la velocità di A è $v_A = 10\ \frac{km}{h}$ mentre quella di B è $v_B = 5\ \frac{km}{h}$.

Una mosca parte insieme ad A e vola verso B; quando lo incontra ritorna verso A; dopo aver raggiunto A torna ancora verso B, fino a che questi si incontrano. Sapendo che la mosca vola a $v_{mosca} = 35\ \frac{km}{h}$, simula il volo della mosca.

Problema 18. Il superenalotto:

ULTIMA ESTRAZIONE DEL 03/02/2024

RUOTE	1° Estratto	2° Estratto	3° Estratto	4° Estratto	5° Estratto
BARI	42	63	20	90	47
CAGLIARI	27	23	59	21	55
FIRENZE	34	32	44	66	13
GENOVA	15	24	30	83	78
MILANO	73	26	5	78	17
NAPOLI	45	18	87	69	5
PALERMO	53	5	28	65	58
ROMA	68	83	15	52	86
TORINO	25	86	13	56	42
VENEZIA	31	5	44	88	27
NAZIONALE	13	89	82	59	51

Oggi non funziona più così.
Il gioco consisteva nell'indovinare i numeri di una sestina generata prendendo da sei diverse ruote un numero, con la seguente regola: dalla prima ruota (in figura Bari) prendo il primo numero 42, dalla seconda ruota prendo il primo numero se è diverso da 42 altrimenti prendo il secondo e così via per la terza, quarta, quinta e sesta ruota.
Simulare la probabilità di fare sei.