Ввод [3]:

```python
from PIL import ImageGrab
from IPython.display import display, Image
def ins(ratio=1.0):
    im_data = ImageGrab.grabclipboard()
    new_size = tuple([int(i*ratio) for i in im_data.size])
    thumb = im_data.resize(new_size)
    fn = "temp.PNG"
    thumb.save(fn)
    img = Image(filename=fn)
    display(img)
```

Ввод [6]:

```python
ins(1)
```

Примеры

3. Предположим, что симметричную монету бросают до первого появления герба. Найти вероятность события A, состоящего в том, что будет произведено не более трех бросаний.
Решение. Пространство элементарных событий является множество

$$\Omega = \left\{ \omega_1 = \text{Г}, \omega_2 = \text{РГ}, \ldots, \omega_n = \underbrace{\text{Р}\ldots\text{Р}}_{(n-1)\text{ раз}}\text{Г}, \ldots ; \omega_\infty \right\},$$

где ω_n означает, что герб впервые появится при n-ом бросании монеты, а ω_∞ соответствует той возможности, что герб никогда не появится (в этом случае наш эксперимент продолжается бесконечно долго). Припишем веса $\mathbb{P}(\omega_n) = \frac{1}{2^n}$, а $\mathbb{P}(\omega_\infty) = 0$. Тогда $\sum_{\omega\in\Omega} \mathbb{P}(\omega) = 1$. Событие $A = \{\text{Г}, \text{РГ}, \text{РРГ}\}$. Поэтому
$\mathbb{P}(A) = \frac{1}{2} + \frac{1}{4} + \frac{1}{8}$
Ответ: $\mathbb{P}(A) = \frac{7}{8}$

Ввод [7]:

```python
from scipy.stats import *
import numpy as np
import random
```

Ввод [10]:

```python
R=randint(0,2)
R.rvs()
```

Out[10]: 0

Ввод [4]:

```python
xk = np.arange(2)
pk = (0.9, 0.1)
R1 = rv_discrete(name='custm', values=(xk, pk))
```

Ввод [5]:

```python
xk
```

Out[5]: array([0, 1])

Ввод [4]:
```
R.rvs()
```
Out[4]: 1

Ввод [7]:
```
random.randint(0, 1)
```
Out[7]: 0

Ввод [11]:
```
def count_N():
    res=[]
    while (r := R.rvs()) != 1:
        #print(r)
        res.append(r)
    res.append(r)  # если мы хотим увидеть в конце выпадение "герба"
    return res,len(res)
```

Ввод [19]:
```
count_N()
```
Out[19]: ([0, 0, 0, 1], 4)

Ввод [20]:
```
N=200000
NA=[]
for i in range(1,N+1):
    sample=count_N()[1]
    if sample<=3:
        NA.append(sample)
```

Ввод [21]:
```
PA=len(NA)/N
PA
```
Out[21]: 0.87578

Ввод [22]:
```
p=1/2
X=geom(p)
```

Ввод [23]:
```
P_A=X.pmf(1)+X.pmf(2)+X.pmf(3)
P_A
```
Out[23]: 0.875

Ввод [24]:
```
N=20
sample=X.rvs(size=N)
sample
```
Out[24]:
```
array([1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 5, 4, 2, 2, 5, 4, 1, 2],
      dtype=int64)
```

Ввод [25]:
```
sample <=3
```
Out[25]:
```
array([ True,  True,  True,  True,  True,  True,  True,  True,  True,
        True,  True,  True, False, False,  True,  True, False, False,
        True,  True])
```

Ввод [26]:
```
sample[sample <=3]
```
Out[26]:
```
array([1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2], dtype=int64)
```

Ввод [27]:
```python
N_A=len(sample[sample <=3])
N_A
```
Out[27]: 16

Ввод [28]:
```python
pstat=N_A/N
pstat
```
Out[28]: 0.8

Ввод [29]:
```python
N=1000000
sample=X.rvs(size=N)
N_A=len(sample[sample <=3])
pstat=N_A/N
pstat
```
Out[29]: 0.875337

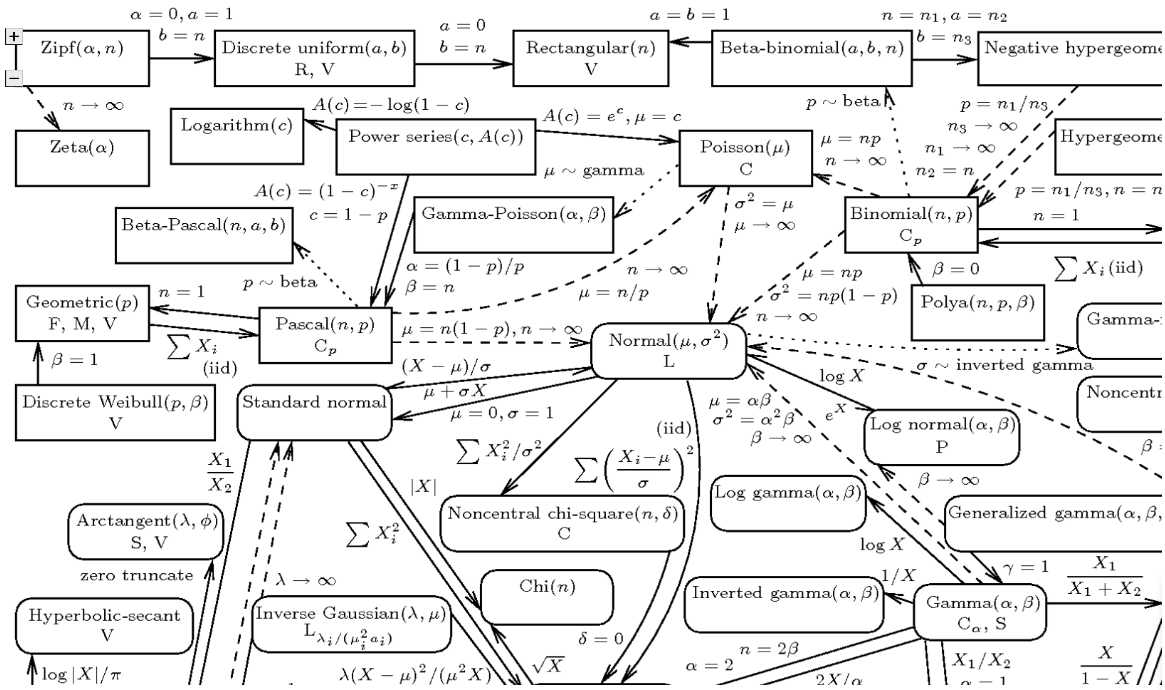
Ввод [31]:
```python
import scipy

```

Ввод [32]:
```python
scipy.version.full_version
```
Out[32]: '1.12.0'

[https://www.math.wm.edu/%7Eleemis/chart/UDR/UDR.html (https://www.math.wm.edu/%7Eleemis/chart/UDR/UDR.html)](https://www.math.wm.edu/%7Eleemis/chart/UDR/UDR.html)

Ввод [27]:
```python
ins(1)
```



Ввод [ ]:
```python

```