```
Ввод [3]:
            1 | from PIL import ImageGrab
            2 | from IPython.display import display, Image
            3 def ins(ratio=1.0):
                 im_data = ImageGrab.grabclipboard()
            4
            5
                   new_size = tuple([int(i*ratio) for i in im_data.size])
            6
                  thumb = im_data.resize(new_size)
            7
                  fn = "temp.PNG"
            8
                   thumb.save(fn)
            9
                   img = Image(filename=fn)
           10
                   display(img)
```

```
Ввод [6]: 1 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

$$\Omega = \left\{ \omega_1 = \Gamma, \omega_2 = P\Gamma, \dots, \omega_n = \underbrace{P \dots P\Gamma}_{(n-1) \text{ pas}}, \dots; \omega_{\infty} \right\},\,$$

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 $A = \{\Gamma, \Pr, \Pr\}$. Поэтому $\mathbb{P}(A) = \frac{1}{2} + \frac{1}{4} + \frac{1}{8}$ Ответ: $\mathbb{P}(A) = \frac{7}{8}$

```
BBOQ [7]: 1 from scipy.stats import *
2 import numpy as np
3 import random

BBOQ [10]: 1 R=randint(0,2)
2 R.rvs()

Out[10]: 0

BBOQ [4]: 1 xk = np.arange(2)
2 pk = (0.9, 0.1)
3 R1 = rv_discrete(name='custm', values=(xk, pk))

BBOQ [5]: 1 xk

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

```
Ввод [4]:
            1 R.rvs()
  Out[4]: 1
 Ввод [7]:
               random.randint(0, 1)
  Out[7]: 0
               def count_N():
Ввод [11]:
            1
             2
                   res=[]
             3
                   while (r := R.rvs()) != 1:
             4
                       #print(r)
             5
                       res.append(r)
                   res.append(r) # если мы хотим увидеть в конце выпадение "герба"
             6
             7
                   return res,len(res)
Ввод [19]:
            1 count_N()
 Out[19]: ([0, 0, 0, 1], 4)
Ввод [20]:
             1 N=200000
             2
              NA=[]
             3
               for i in range(1,N+1):
                   sample=count_N()[1]
             5
                   if sample<=3:</pre>
                       NA.append(sample)
             6
Ввод [21]:
             1 PA=len(NA)/N
             2 PA
 Out[21]: 0.87578
Ввод [22]:
             1 p=1/2
             2 X=geom(p)
Ввод [23]:
            1 P_A=X.pmf(1)+X.pmf(2)+X.pmf(3)
             2 P_A
 Out[23]: 0.875
Ввод [24]:
             1 N=20
             2 sample=X.rvs(size=N)
             3 sample
 Out[24]: array([1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 5, 4, 2, 2, 5, 4, 1, 2],
                 dtype=int64)
Ввод [25]:
            1 sample <=3
 Out[25]: array([ True,
                          True, True, True, True, True, True, True,
                   True,
                          True, True, False, False, True, False, False,
                   True,
                          True])
Ввод [26]:
               sample[sample <=3]</pre>
 Out[26]: array([1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2], dtype=int64)
```

```
Ввод [27]:
                      1 N_A=len(sample[sample <=3])</pre>
   Out[27]: 16
Ввод [28]:
                      1
                           pstat=N_A/N
                           pstat
   Out[28]: 0.8
Ввод [29]:
                      1
                           N=1000000
                           sample=X.rvs(size=N)
                      2
                           N_A=len(sample[sample <=3])</pre>
                      3
                           pstat=N_A/N
                           pstat
   Out[29]: 0.875337
Ввод [31]:
                      1
                           import scipy
Ввод [32]:
                      1 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)
Ввод [27]:
                      1 ins(1)
                                   \alpha=0, a=1
                                                                                                     a=b=1
                                                                                                                                  n=n_1,a=n_2
                                                                          a = 0
                                       b = n Discrete uniform(a, b)
                                                                                                                                      b = n_3 Negative hypergeome
                        \mathrm{Zipf}(\alpha,n)
                                                                                     Rectangular(n)
                                                                                                             Beta-binomial(a, b, n)
                                                        R, V
                                                                                                                        p\sim \mathrm{beta} \hbar
                                                           A(c) = -\log(1-c)
                           n \to \infty
                                                                                        A(c) = e^c, \mu
                                         Logarithm(c)
                                                                                                                                                       Hypergeome
                                                              Power series (c, A(c))
                                                                                                           Poisson(\mu)
                         Zeta(\alpha)
                                                                                                                          n \to \infty
                                                                                                                C
                                                                                                                                                p = \overline{n_1/n_3, n = n}
                                                   A(c) = (1 - c)^{r}
                                                                                                              \sigma^2 = \mu
                                                                                                                             Binomial(n, p)
                                                                        Gamma-Poisson(\alpha, \beta)
                                                         c = 1 -
                                                                                                                                                     n = 1
                                  Beta-Pascal(n, a, b)
                                                                                                                  \mu = np
                                                                                                                                                        \sum X_i (iid)
                                                                      \alpha = (1-p)/p
                                                                                                                       = np(1-p)
                       \overline{\text{Geometric}(p)}
                                                                                                                                      Polya(n, p, \beta)
                                                                                                                  n 	o \infty
                         F, M, V
                                                                                                                                                           Gamma-
                                                      Pascal(n, p)
                                                                     \mu = n(1-p), n \rightarrow \infty
                                                                                               Normal(\mu, \sigma^2)
                         \beta = 1
                                            (iid)
                                                                     (X-\mu)/\sigma
                                                                                                      L
                                                                                                                                          inverted gamm
                                                                                                            \mu = \alpha \beta
\sigma^2 = \alpha^2 \beta
                                                                                                                                                           Noncentr
                                                                        \mu + \sigma X
                      Discrete Weibull(p, \beta)
                                                 Standard normal
                                                                            \mu = 0, \sigma = 1
                                                                                                                                Log normal(\alpha, \beta)
                                                                                                     (iid)
                                                                                                                                        Р
                                                                            \sum X_i^2/\sigma
                                                                                               \left(\frac{X_i - \mu}{2}\right)
                                             \overline{X_2}
                                                                                                   \sigma
                                                                                                                                          Generalized gamma(\alpha, \beta,
                                                                           Noncentral chi-square(n,\delta)
                             Arctangent(\lambda, \phi)
                                                              \sum X_i^2
                                   S, V
                                                                                                                                                            X_1
                                                                                    Chi(n)
                                                                                                          Inverted gamma(\alpha, \beta)
                                                                                                                                                         \overline{X_1 + X_2}
                                                                                                                                       Gamma(\alpha, \beta)
                                                   Inverse Gaussian(\lambda, \mu)
L_{\lambda_i/(\mu_i^2 a_i)}
                       Hyperbolic-secant
                                                                                                                                           C_{\alpha}, S
                                                                                                                n = 2\beta
                                                                                                                                                             X
                                                                                                                                          X_1/X_2
                       \Lambda \log |X|/\pi
                                                             \lambda(X-\mu)^2/(\mu^2X)
                                                                                                                                                           \overline{1-X} //
 Ввод [ ]:
```