

```
In [14]: import math
import itertools
```

```
In [19]: # расчет числа сочетаний без повторений
def calc_combination(n, k):
    return math.factorial(n) / (math.factorial(n - k) * math.factorial(k))

# расчет числа размещений с повторениями
def calc_accommodation(n, k):
    return n**k

# расчет числа перестановок без повторений
def calc_permutation(n):
    return math.factorial(n)
```

1 Задание (теорема сложения)

Найти вероятность выпадения 2 или 5 очков при подбрасывании игральной кости, на гранях которой имеются соответственно 1,2,3,4,5 и 6 очков.

$$A - \text{выпадение двух очков} \quad P(A) = \frac{1}{6}$$

$$B - \text{выпадение пяти очков} \quad P(B) = \frac{1}{6}$$

$$P(A + B) = \frac{1}{6} + \frac{1}{6} = \frac{1}{3}$$

2 Задание (теорема умножения)

Найти вероятность того, что при двух подбрасываниях той же самой игральной кости сначала выпадет 2, а затем 5.

$$P(AB) = \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}$$

3 Задание

Найти вероятность выпадения 2 и 5 очков при двух подбрасываниях той же самой игральной игральной кости. Обратите внимание на порядок выпадения костей!

Не вижу разницы между второй и третьей задачей.

```
In [114... import random
n = 1000
s = 0
for i in range(n):
    k = 0
    a = [random.randint(1, 6) for i in range(n)]
    b = [random.randint(1, 6) for i in range(n)]
    for ai, bi in zip(a, b):
        # print(ai, bi, end=' ')
        if ai == 2 and bi == 5:
            k += 1
    s += k/n
print(n, k, k/n)
print(s/n)

1000 29 0.029
0.0279099999999999963
```

```
In [113... 1/36
```

```
Out[113... 0.027777777777777776
```

4 Задание (Геометрическая вероятность + интервалы)

На отрезке АВ длиной 20 см наугад отметили точку С. Какова вероятность, что она находится на расстоянии не более 9 см от точки А и не более 15 см от точки В?

```
In [12]: AB = 20; AM = 9; NB = 15
MN = AM + NB - AB
p = MN / AB
print('AC <= 9см, BC <= 15см с вероятностью',p)

AC <= 9см, BC <= 15см с вероятностью 0.2
```

5 Задание

Телефонный номер состоит из 7 цифр. Какова вероятность, что это номер 8882227?

```
In [13]: n = 10; k = 7
print('Вероятность, что это номер 8882227:', 1 / calc_accommodation(n, k))

Вероятность, что это номер 8882227: 1e-07
```

6 Задание

Набирая номер телефона, абонент забыл 2 последние цифры, и, помня только то, что эти цифры различны и среди них нет нуля, стал набирать их наудачу. Сколько вариантов ему надо перебрать, чтобы наверняка найти нужный номер? Какова вероятность того, что он угадает номер с первого раза?

```
In [15]: for p in itertools.product("123456789", repeat=2):  
         print(''.join(p), end=' ')
```

```
11 12 13 14 15 16 17 18 19 21 22 23 24 25 26 27 28 29 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 47 48 49 51 52  
53 54 55 56 57 58 59 61 62 63 64 65 66 67 68 69 71 72 73 74 75 76 77 78 79 81 82 83 84 85 86 87 88 89 91 92 93 94  
95 96 97 98 99
```

```
In [16]: n = 9; k = 2  
acc = calc_accomodation(n, k)  
print(f'Число вариантов для перебора: {acc}')  
print(f'Вероятность угадать номер с первого раза:', 1 / acc)
```

Число вариантов для перебора: 81

Вероятность угадать номер с первого раза: 0.012345679012345678

7 Задание** (необязательное)

Чёрный куб покрасили снаружи белой краской, затем разрезали на 27 одинаковых маленьких кубиков и как попало сложили из них большой куб. С какой вероятностью все грани этого куба будут белыми?

Все возможные варианты = $27!24^{27}$

Покрашена одна грань - 6 кубиков, 4 варианта расположения, $P_1 = \frac{6!4^6}{27!24^{27}}$

Покрашено две грани - 12 кубиков, 2 варианта расположения, $P_2 = \frac{12!2^{12}}{27!24^{27}}$

Покрашено три грани - 8 кубиков, 3 варианта расположения, $P_3 = \frac{8!3^8}{27!24^{27}}$

Не покрашено ни одной - 1 черный кубик, 24 варианта(все варианты вращения), $P_4 = \frac{1!24}{27!24^{27}}$

Вероятность события, что все грани этого куба будут белыми: $P_1 \cdot P_2 \cdot P_3 \cdot P_4 = \frac{6! \cdot 4^6 \cdot 12! \cdot 2^{12} \cdot 8! \cdot 3^8 \cdot 24}{27! \cdot 24^{27}} = \frac{8! \cdot 3^8 \cdot 12! \cdot 2^{24} \cdot 6!}{27! \cdot 24^{26}}$

```
In [43]: total = calc_permutation(27)*24**27  
  
varss = calc_accomodation(6)*4**6*calc_accomodation(12)*2**12*calc_accomodation(8)*3**8*24  
  
print('Вероятность, что все грани куба белые:', (varss/total))
```

Вероятность, что все грани куба белые: 1.8298051356415021e-37