

1. Составить список из заглавных букв английского алфавита (в кодировке Unicode), посчитать среднее значение кодов.

```
list1 = CharacterRange["A", "Z"];  
code = ToCharacterCode[list1];  
N[Total[code] / Length[code]]
```

Out[ ]= { 77.5 }

2. Дано число 358138 [number], найти его обратное число [res1] в конечном поле GF(6502301) [GF(div)].

```
number = 358138;  
GF = 6502301;  
answ = PowerMod[number, -1, GF]  
Mod[answ * number, GF]
```

Out[ ]= 4060720

Out[ ]= 1

3. Дано число [start], найти два ближайших к нему простых числа (b1, b2). Посчитать их сумму по модулю [mod].

(Если два ближайших числа, больших start, то параметры NextPrime +1, +2)

```
start = 4246;  
mod = 147;  
b1 = NextPrime[start, -1]  
b2 = NextPrime[start, 1]  
Mod[b1 + b2, mod]
```

Out[ ]= 4243

Out[ ]= 4253

Out[ ]= 117

4. Найти количество разрядов в двоичной записи шестнадцатеричного числа 33A855 [number].

```
number = 16^33A855;  
BaseForm[number, 2]  
IntegerLength[number, 2]
```

Out[ ]//BaseForm=

1100111010100001010101<sub>2</sub>

Out[ ]= 22

5. Найти количество простых чисел в диапазоне от 17 [R1] до 29 [R2].

```

R1 = 17;
R2 = 29;
primeRange [min_, max_] := Module[{list = {}},
If[PrimeQ[min], newNum = min, newNum = NextPrime [min, 1]];
While[newNum ≤ max,
AppendTo [list, newNum];
newNum = NextPrime [newNum, 1]];
list
];
list5 = primeRange[R1, R2]
Length[list5]

```

Out[ ]:= {17, 19, 23, 29}

Out[ ]:= 4

6. Получить множество целых чисел от [min1] до [max1] и множество простых чисел от [min2] до [max2]. Найти произведение элементов пересечения этих множеств.

```

min1 = 65;
max1 = 105;
min2 = 80;
max2 = 145;
listInteger = Range[min1, max1]
listPrime = primeRange[min2, max2]
listIntersection = Intersection[listInteger, listPrime]
mult = 1;
Do[mult = mult * listIntersection[[i]], {i, 1, Length[listIntersection]}];
mult

```

Out[ ]:= {65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105}

Out[ ]:= {83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139}

Out[ ]:= {83, 89, 97, 101, 103}

Out[ ]:= 7 454 155 217

7. Составить список из нечетных чисел в диапазоне от nmin до nmax. Сделать квадратную матрицу. Поменять местами строки str1 и str2. Циклично сдвинуть строку str3 вправо (влево) на len1. Поменять местами столбцы col1 и col2. Циклично сдвинуть столбец col3 вверх (вниз) на len2. Посчитать сумму элементов главной диагонали.  
(Получить список из первых count простых чисел)

7.1. Составить список (нечётные числа в диапазоне от nmin до nmax):

```

In[ ]:= nmin = 1;
nmax = 31;

```

```
list7 = {};
Do[If[Mod[i, 2] > 0, AppendTo[list7, i]], {i, nmin, nmax}];
list7
```

```
Out[ ]:= {1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31}
```

7.1. Составить список (первые count простых чисел):

```
In[ ]:= count = 1764;
list7 = {};
Do[AppendTo[list7, NextPrime[1, i]], {i, 1, count}];
list7
```

7.2. Сделать квадратную матрицу: (если что-то пошло не так, возвращаемся к этому пункту и следующие действия по очереди)

```
In[ ]:= mA7 = Partition[list7, Sqrt[Length[list7]]] // MatrixForm;
```

7.3. Поменять местами строки str1 и str2:

```
In[ ]:= str1 = 30;
str2 = 6;
vr = mA7[[1, str1]];
mA7[[1, str1]] = mA7[[1, str2]];
mA7[[1, str2]] = vr;
mA7;
```

7.4. Циклически сдвинуть строку str3 на len1 позиций вправо (RotateRight) / влево (RotateLeft):

```
str3 = 33;
len1 = 41;
mA7[[1, str3]] = RotateRight[mA7[[1, str3]], len1];
mA7;
```

7.5. Поменять местами столбцы col1 и col2:

```
In[ ]:= col1 = 4;
col2 = 10;
vrcol = {};
Do[AppendTo[vrcol, mA7[[1, i, col1]]], {i, 1, Length[mA7[[1]]] }];
Do[Part[mA7, 1, i, col1] = Part[mA7, 1, i, col2], {i, 1, Length[mA7[[1]]] }];
Do[Part[mA7, 1, i, col2] = vrcol[[i]], {i, 1, Length[mA7[[1]]] }];
mA7;
```

7.6. Циклически сдвинуть столбец col3 на len2 позиций вниз (RotateRight) / вверх (RotateLeft):

```
col3 = 3;
len2 = 90;
vrcol = {};
Do[AppendTo[vrcol, mA7[[1, i, col3]]], {i, 1, Length[mA7[[1]]] }];
vrcol = RotateRight[vrcol, len2];
Do[Part[mA7, 1, i, col3] = vrcol[[i]], {i, 1, Length[mA7[[1]]] }];
mA7;
```

7.7. Рассчитать сумму элементов главной диагонали:

```
sum = 0;
Do[sum = sum + Part[mA7, 1, i, i], {i, 1, Length[mA7[[1]]}];
sum
```

Out[8]= 309 871

8. Установить генератор случайных чисел в начальное состояние с параметром (param)  $2^{15} \pmod{311}$ . Получить список из 10 случайных простых чисел в диапазоне от 46 до 77. Найти произведение двух простых чисел, встречающихся в списке с максимальной (maxfr) и минимальной (minfr) частотами. В случае наличия чисел с одинаковыми частотами, выбирать первые в списке.

```
param = Mod[2 ^ 15, 463];
min = 75 000;
max = 121 000;
count = 10 000;
SeedRandom[param];
list8 = RandomPrime[{min, max}, count];
list8 = Tally[list8];
listMAX = {};
Do[If[list8[[i, 2]] == Max[list8[[All, 2]]], AppendTo[listMAX, list8[[i, 1]]],
  {i, 1, Length[list8]}];
listMAX;
listMIN = {};
Do[If[list8[[i, 2]] == Min[list8[[All, 2]]], AppendTo[listMIN, list8[[i, 1]]],
  {i, 1, Length[list8]}];
listMIN;
listMAX[[1]] * listMIN[[1]]
```

Out[9]= 11 297 452 321

9. Найти значение функции Эйлера для числа  $x$ , которое определяется из соотношения:  $a \cdot x + b = c \pmod{n}$ , где  $a = 34\,535$ ,  $b = 34\,745$ ,  $c = 26\,341$ ,  $n = 11\,047$ .

```
a = 34 535;
b = 34 745;
c = 26 341;
n = 11 047;
list9 = Range[0, n - 1];
answ = {};
Do[If[Mod[a * list9[[i]] + b, n] == Mod[c, n], AppendTo[answ, list9[[i]]],
  {i, 1, Length[list9]}];
answ
EulerPhi[answ[[1]]]
```

Out[10]= {517}

Out[11]= 460

10. Определить ожидаемое время раскрытия пароля длиной (length) символов и содержащего следующие наборы: {цифры, строчные русские, строчные латинские, прописные латинские}, если скорость перебора пароля (в символах в секунду) ( $V$ ) равна

обратному элементу числа 1939 по модулю 661. Ответ вводить как целое число суток.

Длины наборов:

```
Length[CharacterRange["a", "я"]]
```

```
Length[CharacterRange["a", "z"]]
```

```
Length[Range[0, 9]]
```

```
Out[8]= 32
```

```
Out[9]= 26
```

```
Out[10]= 10
```

```
v = PowerMod [2825, - 1, 503];
```

```
length = 8;
```

```
alf = 32 + 10;
```

```
IntegerPart[Divide[(alf^length), (2 * v * 60 * 60 * 24)]]
```

```
Out[11]= 651 556
```

11. Архив текстового файла archive-118.zip защищен паролем из 4 символов, содержащих строчные и заглавные латинские буквы, а также все цифры. Один из символов пароля можно определить из следующего условия: полусумма кода символа и кода позиции символа в пароле равна 69, полуразность кода символа и кода позиции символа равна 20. Исключить пробелы и подсчитать число символов в тексте.

```
Solve[(code + pos) / 2 == 69 && (code - pos) / 2 == 20, {code, pos}]
```

```
Out[12]= {{code -> 89, pos -> 49}}
```

```
FromCharacterCode[89]
```

```
FromCharacterCode[49]
```

```
Out[13]= Y
```

```
Out[14]= 1
```

```

text11 =
  "умеренности по зеленой влаге каналов скользили лодки изящные как бронзовые
  цветки в нескончаемо длинных рядах жилищ извивающихся по склонам подобно
  оцепеневшим змеям в прохладных ночных постелях лениво перешептывались
  возлюбленные под факелами на аллеях держа в руках извергающих тончайшую
  паутину золотых пауков еще бегали заигравшиеся детитут и там на столах
  булькающих серебристой лавой готовился поздний ужин вампитеатрах
  сотен городов на ночной стороне марса смуглые марсиане с глазами";
text11 = StringReplace[text11, " " → ""]
StringLength[text11]

```

Out[4]= 425

умеренности по зеленой влаге каналов скользили лодки изящные как бронзовые цветки в нескончаемо длинных рядах жилищ извивающихся по склонам подобно оцепеневшим змеям в прохладных ночных постелях лениво перешептывались возлюбленные под факелами на аллеях держа в руках извергающих тончайшую паутину золотых пауков еще бегали заигравшиеся детитут и там на столах булькающих серебристой лавой готовился поздний ужин вампитеатрах сотен городов на ночной стороне марса смуглые марсиане с глазами

Out[4]= 425

12. Скачайте с сетевого диска (ftp-сервера) файл text-02.txt, расположенный в папке Texts и определите энтропию сообщения, содержащегося в нем. Ответ представить в битах, с 5 знаками после запятой N[..., 6].

```

str12 = "окраину дремлющего мертвого города озаренного
светом двух догоняющих другд друга лун тени протянувшиеся
от их ног были двойными несколько минуткосмонавты стояли
затаив дыхание ждали вот сейчас чтонибуть шевельнетсяв этом
безжизненном городе возникнет какойнибуть туманный силуэт";
list12 =
StringSplit["окраину дремлющего мертвого города озаренного светом двух
догоняющих другд друга лун тени протянувшиеся от их ног были
двойными несколько минуткосмонавты стояли затаив дыхание
ждали вот сейчас чтонибуть шевельнетсяв этом безжизненном
городе возникнет какойнибуть туманный силуэт", ""]
list12T = Tally[list12]
p12 = N[list12T[[All, 2]] / Length[list12]]
Total[p12]
NumberForm[- Sum[p12[[i]] * Log[2, p12[[i]]], {i, Length[p12]}], 8]
N[Entropy[2, str12], 8]

```

```

Out[ ]= {О, К, Р, А, И, Н, У, , Д, Р, Е, М, Л, Ю, Щ, Е, Г, О, , М, Е, Р, Т, В, О, Г, О, , Г,
О, Р, О, Д, А, , О, З, А, Р, Е, Н, Н, О, Г, О, , С, В, Е, Т, О, М, , Д, В, У, Х,
, Д, О, Г, О, Н, Я, Ю, Щ, И, Х, , Д, Р, У, Г, Д, Р, У, Г, А, , Л, У, Н, , Т, Е,
Н, И, , П, Р, О, Т, Я, Н, У, В, Ш, И, Е, С, Я, , О, Т, , И, Х, , Н, О, Г, , Б,
Ы, Л, И, , Д, В, О, Й, Н, Ы, М, И, , Н, Е, С, К, О, Л, Ъ, К, О, , М, И, Н, У, Т,
К, О, С, М, О, Н, А, В, Т, Ы, , С, Т, О, Я, Л, И, , З, А, Т, А, И, В, , Д, Ы,
Х, А, Н, И, Е, , Ж, Д, А, Л, И, , В, О, Т, , С, Е, Й, Ч, А, С, , Ч, Т, О, Н,
И, Б, У, Д, Ъ, , Ш, Е, В, Е, Л, Ъ, Н, Е, Т, С, Я, В, , Э, Т, О, М, , Б, Е, З,
Ж, И, З, Н, Е, Н, Н, О, М, , Г, О, Р, О, Д, Е, , В, О, З, Н, И, К, Н, Е, Т, ,
К, А, К, О, Й, Н, И, Б, У, Д, Ъ, , Т, У, М, А, Н, Н, Ы, Й, , С, И, Л, У, Э, Т}

```

```

Out[ ]= {{О, 29}, {К, 7}, {Р, 9}, {А, 12}, {И, 17}, {Н, 23}, {У, 11}, { , 33}, {Д, 12}, {Е, 17},
{М, 9}, {Л, 8}, {Ю, 2}, {Щ, 2}, {Г, 9}, {Т, 16}, {В, 11}, {З, 5}, {С, 9}, {Х, 4},
{Я, 5}, {П, 1}, {Ш, 2}, {Б, 4}, {Ы, 5}, {Й, 4}, {Ь, 4}, {Ж, 2}, {Ч, 2}, {Э, 2}}

```

```

Out[ ]= {0.105072, 0.0253623, 0.0326087, 0.0434783, 0.0615942, 0.0833333,
0.0398551, 0.119565, 0.0434783, 0.0615942, 0.0326087, 0.0289855,
0.00724638, 0.00724638, 0.0326087, 0.057971, 0.0398551, 0.0181159,
0.0326087, 0.0144928, 0.0181159, 0.00362319, 0.00724638, 0.0144928,
0.0181159, 0.0144928, 0.0144928, 0.00724638, 0.00724638, 0.00724638}

```

```

Out[ ]= 1.

```

```

Out[ ]//NumberForm=

```

```

4.437849

```

```

Out[ ]= 4.4378490

```

13. Определите энтропию сектора с номером 795 виртуального флоппи-диска flptest.flp с точностью 5 знаков после запятой. Для округления результата применять функцию N[.]. Пример ввода: 5.55555.

```

datfile =
"2 136 117 175 30 148 10 210 31 15 107 110 82 121 224 132 15 174 44 161 35
140 225 42 21 125 71 165 198 163 37 252 125 29 131 154 168 233 150
120 196 118 77 216 60 240 162 31 220 174 36 87 75 158 6 197 206 140
168 16 123 110 178 141 220 210 150 105 168 204 229 241 69 172 122 8
121 91 82 108 91 41 180 239 189 6 58 191 106 3 69 27 126 187 0 34 20
88 180 31 237 248 73 30 162 39 250 93 88 167 216 26 182 55 183 108
29 68 1 228 195 44 148 168 165 131 120 47 205 137 129 144 119 182
101 143 93 242 40 60 4 208 47 253 73 241 230 34 197 57 226 234 172
210 253 209 199 205 123 189 121 56 204 202 18 105 200 103 60 57 78
143 127 154 28 148 65 229 153 182 36 133 80 135 30 125 83 208 146 17
131 211 155 168 153 120 150 220 32 197 2 77 207 155 41 135 148 158
222 193 220 61 8 244 242 122 182 145 110 98 238 47 193 143 49 155
192 179 125 231 112 16 250 209 7 106 89 118 117 52 6 72 224 70 82
80 179 252 33 32 173 204 176 147 1 10 49 51 49 233 103 132 36 156
56 32 115 34 147 189 61 52 134 100 36 218 34 186 170 62 180 160 54
208 99 154 202 158 150 99 29 138 9 115 44 121 250 192 53 216 26 68
26 57 153 129 233 55 76 134 241 74 236 169 125 78 121 175 26 167 87
28 41 89 160 218 42 194 47 32 68 124 216 117 218 46 21 80 14 105 13
155 37 70 183 83 183 159 156 238 192 220 250 206 244 126 99 32 33
210 171 89 186 200 67 169 6 220 115 233 46 238 139 178 80 5 23 58
122 179 72 5 155 225 171 228 163 203 4 25 219 108 236 220 222 95 187
128 117 220 2 203 165 72 106 220 28 0 47 119 184 52 5 69 124 238 141
105 244 75 8 140 153 195 15 67 122 101 69 64 243 5 188 109 1 48 202
231 240 19 117 128 166 232 218 126 151 90 221 12 70 78 203 177 195
237 129 0 155 5 241 247 209 221 69 181 141 182 106 204 245 44 20 165
64 45 55 100 60 106 4 100 3 202 141 198 184 89 80 20 55 88 19 123
245 235 151 7 90 57 157 102 188 28 27 149 47 189 235 126 121 190";
list13 = ReadList[StringToStream[datfile], Number];
fr = Tally[list13];
p = N[fr[[All, 2]] / Length[list13]];
Total[p]
ent = N[- Sum[p[[i]] * Log[2, p[[i]]], {i, Length[p]}], 6]

```

Out[ ]= 1.

Out[ ]= 7.54834

14. При стартовом значении генератора равном start сформировать последовательность, состоящую из count случайных чисел, лежащих в диапазоне [min, max]. Найти произведение элементов последовательности, принадлежащих подмножеству, содержащему **четырёхкратную коллизию**. В поле ответа ввести количество разрядов **для шестнадцатеричного представления** полученного произведения.



```

start = 34;
count = 879;
min = 217;
max = 869;
SeedRandom[start];
list14 = RandomInteger[{min, max}, count];
list14T = Tally[list14];
pr = 1;
For[i = 1, i ≤ Length[list14T], i++,
  If[list14T[[i, 2]] == 5, pr = pr * list14T[[i, 1]]];
pr
Length[IntegerDigits[pr, 16]]

```

Out[8]= 6 074 204 547 751 952 040

Out[9]= 16

15. В поле целых чисел определить сумму элементов приведенной системы вычетов по модулю mod.

```

mod = 30;
list15 = Mod[Range[1, mod], mod];
listpriv = {};
sum = 0;
Do[If[GCD[list15[[i]], mod] == 1, AppendTo[listpriv, list15[[i]]],
  {i, Length[list15]}];
listpriv;
EulerPhi[mod] == Length[listpriv]
Total[listpriv]

```

Out[10]= True

Out[11]= 120

16. В поле GF[313] определить произведение обратного элемента по сложению числа a=241 и обратного элемента по умножению для числа b=106

```

field = 313;
a = 241;
b = 106;
Mod[(field - a) * (PowerMod[b, -1, field]), field]

```

Out[12]= 231

17. Определите количество положительных целых чисел, меньших number, которые взаимно просты с number.

```

number = 5045;
count = 0;
Do[If[CoprimeQ[i, number], count = count + 1], {i, 1, number - 1}];
count

```

Out[13]= 4032

18. Создайте два списка чисел, объедините, отсортируйте, при значении генератора start осуществите перестановку элементов. Определить номера позиций максимального и минимального элемента в списке и найдите их (номеров позиций) произведение.

RandomChoice

```
list181 = Range[1, 30];
list182 = Range[259, 1164];
start = 667 482 789;
list18 = Join[list181, list182];
SeedRandom[start];
list18 = RandomSample[list18];
posmin = Position[list18, Min[list18]]
posmax = Position[list18, Max[list18]]
posmax * posmin
```

Out[8]= {{828}}

Out[9]= {{41}}

Out[10]= {{33 948}}

19. Простые случайные числа, сумма последних n

```
start = 111 432 590;
min = 6678;
max = 73 458;
count = 74;
n = 5;
SeedRandom[start];
list19 = RandomPrime[{min, max}, count];
sum = 0;
Do[sum = sum + list19[[i]], {i, Length[list19] - n + 1, Length[list19]}]
sum
```

Out[11]= 140 849

20. Число простых чисел между

PrimePi[29] - PrimePi[17]

Out[12]= 3