

---

## Hexi—PS11—2025 - 03- 18

### Exercises from EIWL3 Section 29

In[482]:=

```
Array[Prime[#] &, 100]
```

Out[482]=

```
{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79,
83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163,
167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251,
257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349,
353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439,
443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541}
```

In[483]:=

```
Array[Prime[# + 1] - Prime[#] &, 100]
```

Out[483]=

```
{1, 2, 2, 4, 2, 4, 2, 4, 6, 2, 6, 4, 2, 4, 6, 6, 2, 6, 4, 2, 6, 4, 6, 8, 4, 2,
4, 2, 4, 14, 4, 6, 2, 10, 2, 6, 6, 4, 6, 6, 2, 10, 2, 4, 2, 12, 12, 4, 2, 4,
6, 2, 10, 6, 6, 6, 2, 6, 4, 2, 10, 14, 4, 2, 4, 14, 6, 10, 2, 4, 6, 8, 6, 6, 4,
6, 8, 4, 8, 10, 2, 10, 2, 6, 4, 6, 8, 4, 2, 4, 12, 8, 4, 8, 4, 6, 12, 2, 18, 6}
```

In[484]:=

```
Array[Plus, {10, 10}] // Grid
```

Out[484]=

```
2 3 4 5 6 7 8 9 10 11
3 4 5 6 7 8 9 10 11 12
4 5 6 7 8 9 10 11 12 13
5 6 7 8 9 10 11 12 13 14
6 7 8 9 10 11 12 13 14 15
7 8 9 10 11 12 13 14 15 16
8 9 10 11 12 13 14 15 16 17
9 10 11 12 13 14 15 16 17 18
10 11 12 13 14 15 16 17 18 19
11 12 13 14 15 16 17 18 19 20
```

In[485]:=

```
FoldList[Times, 1, Range[10]]
```

Out[485]=

```
{1, 1, 2, 6, 24, 120, 720, 5040, 40320, 362880, 3628800}
```

In[486]:=

```
FoldList[Times, 1, Array[Prime, 10]]
```

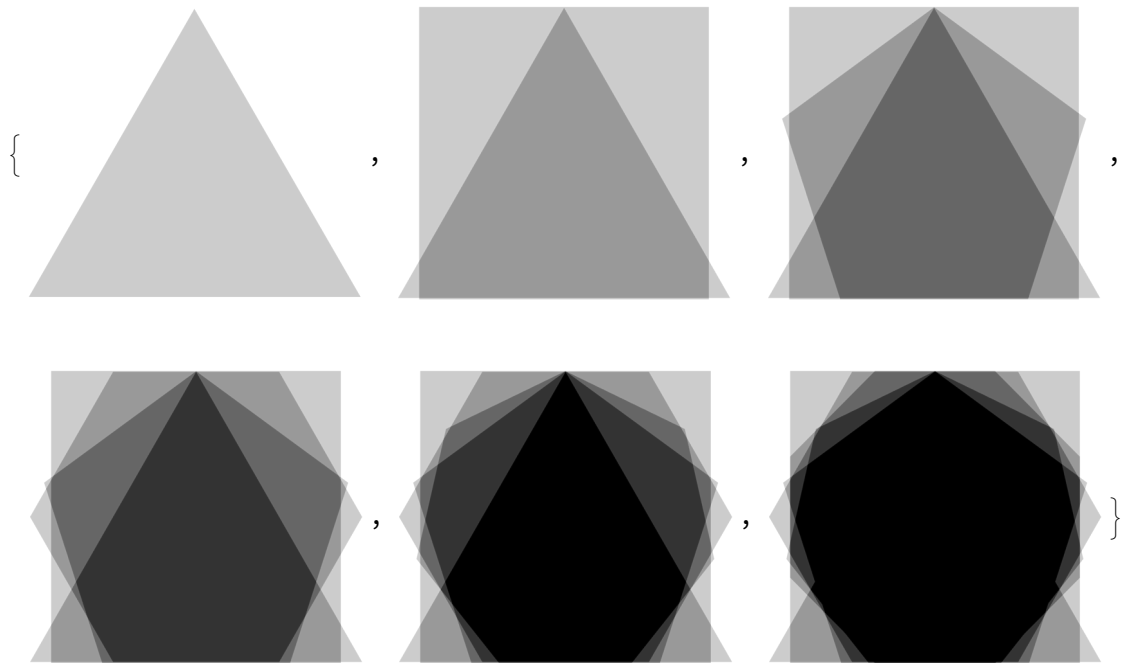
Out[486]=

```
{1, 2, 6, 30, 210, 2310, 30030, 510510, 9699690, 223092870, 6469693230}
```

In[487]:=

```
FoldList[ImageAdd, Graphics[{Opacity[0.2], RegularPolygon[#]}] & /@ Range[3, 8]]
```

Out[487]=



## Exercises from EIWL3 Section 29

In[488]:=

```
Thread[Alphabet[] → LetterNumber[Alphabet[]]]
```

Out[488]=

```
{a → 1, b → 2, c → 3, d → 4, e → 5, f → 6, g → 7, h → 8,
 i → 9, j → 10, k → 11, l → 12, m → 13, n → 14, o → 15, p → 16, q → 17,
 r → 18, s → 19, t → 20, u → 21, v → 22, w → 23, x → 24, y → 25, z → 26}
```

In[489]:=

```
Partition[Take[Alphabet[], 24], 6] // Grid
```

Out[489]=

```
a b c d e f
g h i j k l
m n o p q r
s t u v w x
```

In[490]:=

```
Grid[Partition[IntegerDigits[2^1000], 50], Frame → All]
```

Out[490]=

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 7 | 1 | 5 | 0 | 8 | 6 | 0 | 7 | 1 | 8 | 6 | 2 | 6 | 7 | 3 | 2 | 0 | 9 | 4 | 8 | 4 | 2 | 5 | 0 | 4 | 9 | 0 | 6 | 0 | 0 | 0 | 1 | 8 | 1 | 0 | 5 | 6 | 1 | 4 | 0 | 4 | 8 | 1 | 1 | 7 |
| 3 | 3 | 6 | 0 | 7 | 4 | 4 | 3 | 7 | 5 | 0 | 3 | 8 | 8 | 3 | 7 | 0 | 3 | 5 | 1 | 0 | 5 | 1 | 1 | 2 | 4 | 9 | 3 | 6 | 1 | 2 | 2 | 4 | 9 | 3 | 1 | 9 | 8 | 3 | 7 | 8 | 8 | 1 | 5 | 6 | 9 | 5 |
| 1 | 2 | 7 | 5 | 9 | 4 | 6 | 7 | 2 | 9 | 1 | 7 | 5 | 5 | 3 | 1 | 4 | 6 | 8 | 2 | 5 | 1 | 8 | 7 | 1 | 4 | 5 | 2 | 8 | 5 | 6 | 9 | 2 | 3 | 1 | 4 | 0 | 4 | 3 | 5 | 9 | 8 | 4 | 5 | 7 | 7 | 5 |
| 9 | 8 | 5 | 7 | 4 | 8 | 0 | 3 | 9 | 3 | 4 | 5 | 6 | 7 | 7 | 7 | 4 | 8 | 2 | 4 | 2 | 3 | 0 | 9 | 8 | 5 | 4 | 2 | 1 | 0 | 7 | 4 | 6 | 0 | 5 | 0 | 6 | 2 | 3 | 7 | 1 | 1 | 4 | 1 | 8 | 7 | 7 |
| 1 | 8 | 2 | 1 | 5 | 3 | 0 | 4 | 6 | 4 | 7 | 4 | 9 | 8 | 3 | 5 | 8 | 1 | 9 | 4 | 1 | 2 | 6 | 7 | 3 | 9 | 8 | 7 | 6 | 7 | 5 | 5 | 9 | 1 | 6 | 5 | 5 | 4 | 3 | 9 | 4 | 6 | 0 | 7 | 7 | 0 | 6 |
| 4 | 5 | 7 | 1 | 1 | 9 | 6 | 4 | 7 | 7 | 6 | 8 | 6 | 5 | 4 | 2 | 1 | 6 | 7 | 6 | 6 | 0 | 4 | 2 | 9 | 8 | 3 | 1 | 6 | 5 | 2 | 6 | 2 | 4 | 3 | 8 | 6 | 8 | 3 | 7 | 2 | 0 | 5 | 6 | 6 | 8 | 6 |

In[491]:=

```
Grid[Partition[Take[Characters[WikipediaData["computers"]], 400], 20], Frame → All]
```

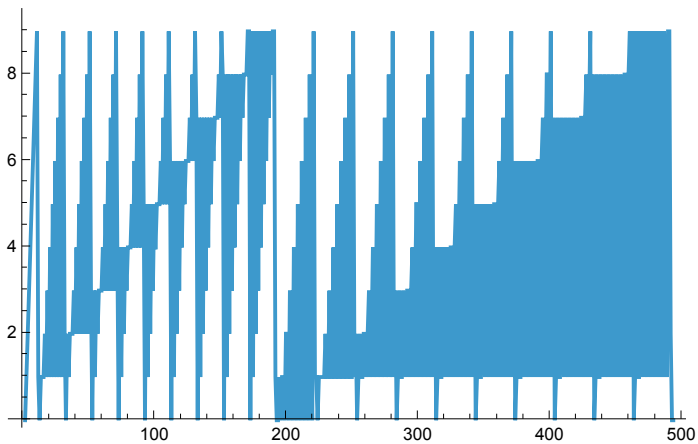
Out[491]=

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | c | o | m | p | u | t | e | r |   | i | s |   | a |   | m | a | c | h |
| i | n | e |   | t | h | a | t |   | c | a | n |   | b | e |   | p | r | o |
| r | a | m | m | e | d |   | t | o |   | a | u | t | o | m | a | t | i | c |
| a | l | l | y |   | c | a | r | r | y |   | o | u | t |   | s | e | q | u |
| e | n | c | e | s |   | o | f |   | a | r | i | t | h | m | e | t | i | c |
|   | o | r |   | l | o | g | i | c | a | l |   | o | p | e | r | a | t | i |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

In[492]:=

```
ListLinePlot[Flatten[IntegerDigits[Range[0, 200]]]]
```

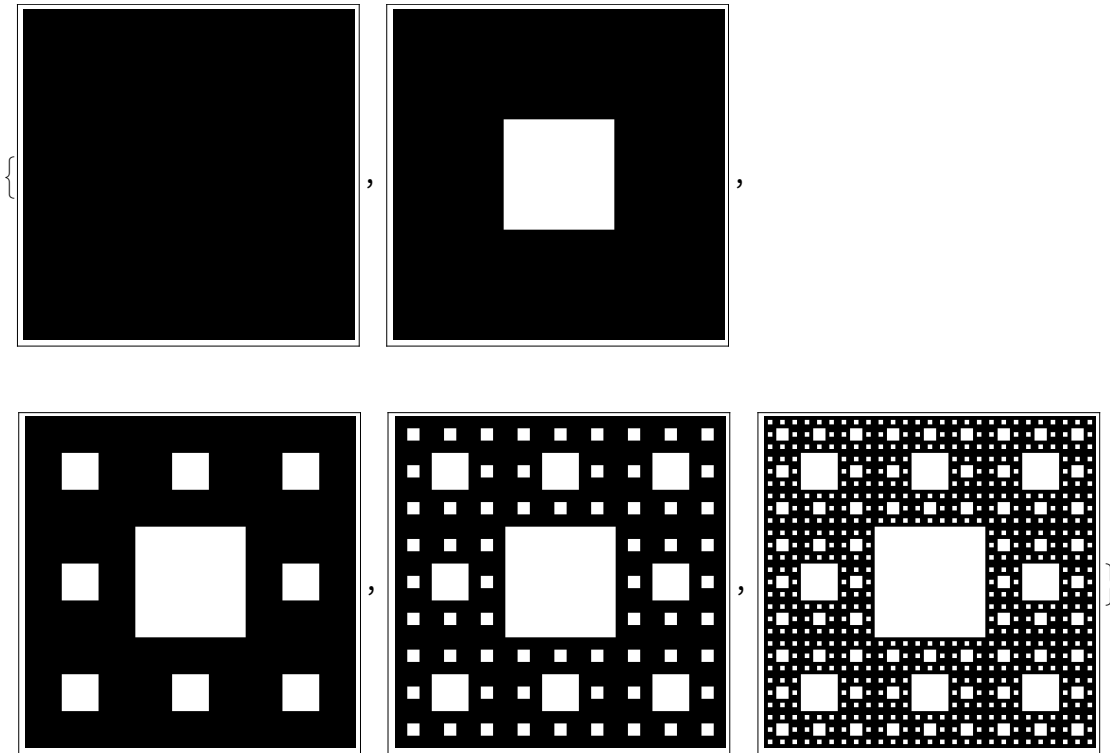
Out[492]=



In[493]:=

```
ArrayPlot /@ NestList[ArrayFlatten[{{#, #, #}, {#, 0, #}, {#, #, #}}] &, {{1}}, 4]
```

Out[493]=



In[494]:=

```
Select[Flatten[Table[{x, y, Sqrt[x^2 + y^2]}, {x, 20}, {y, 20}], 1], IntegerQ[#[[3]]] &]
```

Out[494]=

```
{ {3, 4, 5}, {4, 3, 5}, {5, 12, 13}, {6, 8, 10},  
  {8, 6, 10}, {8, 15, 17}, {9, 12, 15}, {12, 5, 13}, {12, 9, 15},  
  {12, 16, 20}, {15, 8, 17}, {15, 20, 25}, {16, 12, 20}, {20, 15, 25} }
```

In[495]:=

```
Table[Max[Length /@ Split[IntegerDigits[2^n]]], {n, 1, 100}]
```

Out[495]=

```
{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 2, 2, 1, 1, 1, 2, 3,  
  2, 2, 2, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 3, 3, 4, 3, 3, 3, 3, 2, 2, 1, 2,  
  3, 2, 2, 2, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 3, 3,  
  3, 3, 3, 2, 2, 1, 2, 2, 3, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2}
```

In[496]:=

**GatherBy[IntegerName /@ Range[100], StringTake[#, 1] &]**

Out[496]=

```
{ {one, one hundred}, {two, three, ten, twelve, thirteen, twenty, twenty-one,
  twenty-two, twenty-three, twenty-four, twenty-five, twenty-six, twenty-seven,
  twenty-eight, twenty-nine, thirty, thirty-one, thirty-two, thirty-three,
  thirty-four, thirty-five, thirty-six, thirty-seven, thirty-eight, thirty-nine},
  {four, five, fourteen, fifteen, forty, forty-one, forty-two, forty-three,
  forty-four, forty-five, forty-six, forty-seven, forty-eight,
  forty-nine, fifty, fifty-one, fifty-two, fifty-three, fifty-four,
  fifty-five, fifty-six, fifty-seven, fifty-eight, fifty-nine},
  {six, seven, sixteen, seventeen, sixty, sixty-one, sixty-two, sixty-three,
  sixty-four, sixty-five, sixty-six, sixty-seven, sixty-eight, sixty-nine,
  seventy, seventy-one, seventy-two, seventy-three, seventy-four,
  seventy-five, seventy-six, seventy-seven, seventy-eight, seventy-nine},
  {eight, eleven, eighteen, eighty, eighty-one, eighty-two, eighty-three,
  eighty-four, eighty-five, eighty-six, eighty-seven, eighty-eight, eighty-nine},
  {nine, nineteen, ninety, ninety-one, ninety-two, ninety-three, ninety-four,
  ninety-five, ninety-six, ninety-seven, ninety-eight, ninety-nine} }
```

In[497]:=

**SortBy[Take[WordList[], 50], StringTake[#, -1] &]**

Out[497]=

```
{a, abandoned, abashed, abbreviated, abed, abalone, abase, abate, abbe, abbreviate,
  abdicate, abeyance, abhorrence, abidance, abide, abducting, abiding, aah,
  abash, aardvark, aback, abdominal, abeam, abandon, abbreviation, abdication,
  abdomen, abduction, aberration, abjection, abattoir, abductor, abettor,
  abhor, abacus, abbess, abaft, abandonment, abasement, abashment, abatement,
  abbot, abduct, aberrant, abet, abhorrent, abject, abbey, ability, abjectly}
```

In[498]:=

**SortBy[#^2 & /@ Range[20], IntegerDigits[#][[1] &]**

Out[498]=

```
{1, 16, 100, 121, 144, 169, 196, 25, 225, 256, 289, 36, 324, 361, 4, 49, 400, 64, 81, 9}
```

In[499]:=

**SortBy[Range[20], StringLength[IntegerName[#]] &]**

Out[499]=

```
{1, 2, 6, 10, 4, 5, 9, 3, 7, 8, 11, 12, 20, 15, 16, 13, 14, 18, 19, 17}
```

In[500]:=

**GatherBy[RandomSample[WordList[], 20], StringLength]**

Out[500]=

```
{{scarce, nutmeg, abrupt}, {thimbleful, excrescent}, {pancreas, yourself, whipcord},
  {entrancement}, {conflagration, insidiousness, heterogeneity, loathsomeness},
  {monotonic, strongman}, {longsighted},
  {compressibility, totalitarianism}, {thaw, whew}}
```







```
In[501]:= Complement[Alphabet["Ukrainian"], Alphabet["Russian"]]
```

```
Out[501]= {є, і, ї, ґ}
```

```
In[502]:= Intersection[#^2 & /@ Range[100], #^3 & /@ Range[100]]
```

```
Out[502]= {1, 64, 729, 4096}
```

```
In[503]:= Intersection[EntityList[ North Atlantic Treaty Organization COUNTRIES],  
EntityList[ Group of 8 COUNTRIES]]
```

```
Out[503]= { Canada,  France,  Germany,  Italy,  United Kingdom,  United States}
```

```
In[504]:= Transpose[Permutations[Range[1, 4]]] // Grid
```

```
Out[504]= 

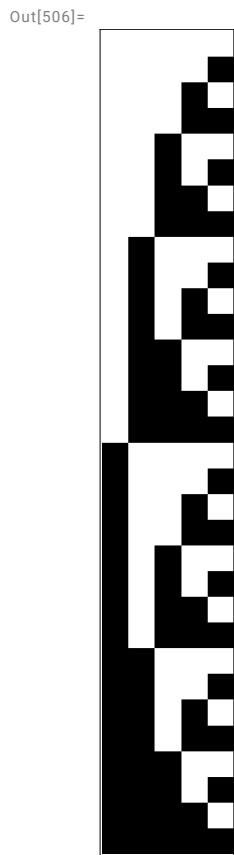
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 |
| 2 | 2 | 3 | 3 | 4 | 4 | 1 | 1 | 3 | 3 | 4 | 4 | 1 | 1 | 2 | 2 | 4 | 4 | 1 | 1 | 2 | 2 | 3 | 3 |
| 3 | 4 | 2 | 4 | 2 | 3 | 3 | 4 | 1 | 4 | 1 | 3 | 2 | 4 | 1 | 4 | 1 | 2 | 2 | 3 | 1 | 3 | 1 | 2 |
| 4 | 3 | 4 | 2 | 3 | 2 | 4 | 3 | 4 | 1 | 3 | 1 | 4 | 2 | 4 | 1 | 2 | 1 | 3 | 2 | 3 | 1 | 2 | 1 |


```

```
In[505]:= StringJoin /@ Permutations[Characters["hello"]]
```

```
Out[505]= {hello, helol, heoll, hlelo, hleol, hlleo, hlloe, hloel, hlle, hoell, holel, holle,  
ehllo, ehlol, eholl, elhlo, elhol, ellho, elloh, elohl, elolh, eohll, eolhl, eollh,  
lhelo, lheol, lhleo, lhloe, lhoel, hlole, lehlo, lehol, lelho, leloh, leohl, leolh,  
llheo, llhoe, lleho, lleoh, llohe, lloeh, lohel, lohle, loehl, loelh, lolhe, loleh,  
ohell, ohlel, ohlle, oehll, oelhl, oeljh, olhel, olhle, olehl, olelh, ollehe, olleh}
```

```
In[506]:=
ArrayPlot[Tuples[{0, 1}, 5]]
```



```
In[507]:=
Table[StringJoin[RandomChoice[Alphabet[], 5]], 10]
```

```
Out[507]=
{gclcq, moqse, kupub, paium, owktz, bkwzs, ipcgw, asrnm, sepuk, iexdn}
```

```
In[508]:=
Flatten[Table[{i, j, k}, {i, 2}, {j, 2}, {k, 2}], 2]
Tuples[Range[2], 3]
```

```
Out[508]=
{{1, 1, 1}, {1, 1, 2}, {1, 2, 1}, {1, 2, 2}, {2, 1, 1}, {2, 1, 2}, {2, 2, 1}, {2, 2, 2}}
```

```
Out[509]=
{{1, 1, 1}, {1, 1, 2}, {1, 2, 1}, {1, 2, 2}, {2, 1, 1}, {2, 1, 2}, {2, 2, 1}, {2, 2, 2}}
```