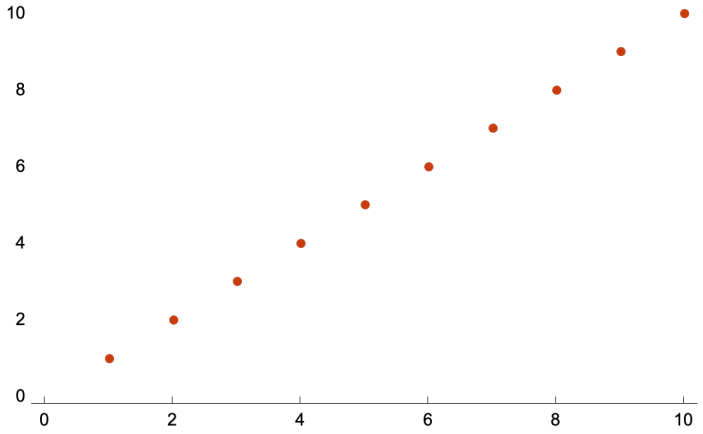
Jeremy, Looks great! 10/10
Watch out for off-by-one errors. See p. 9.

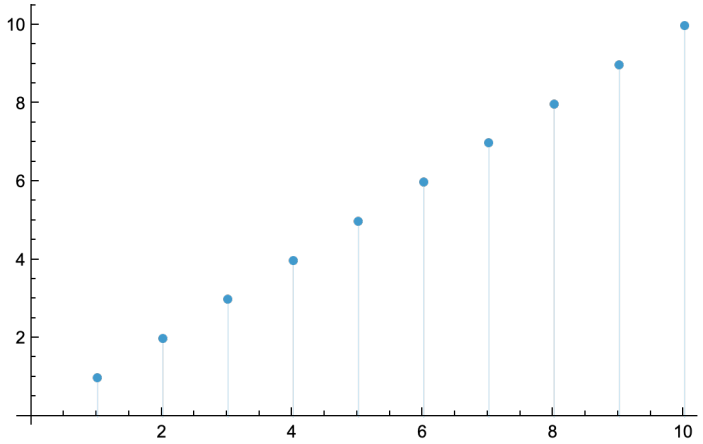
In[1]:=
```
ListPlot[Range[10], PlotTheme → "Web"]
```

Out[1]=



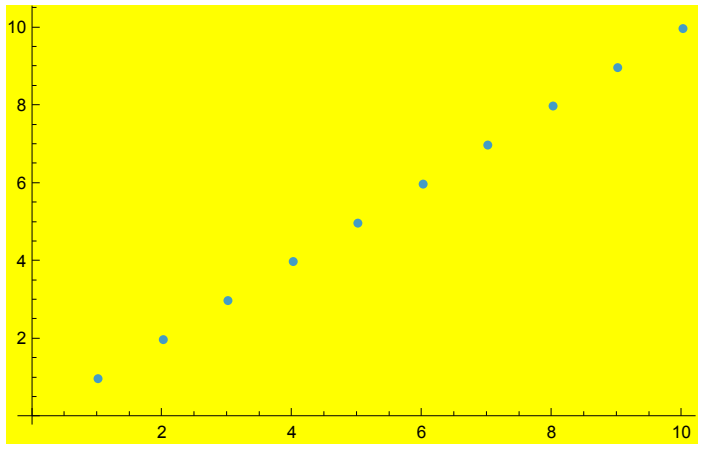
In[2]:=
```
ListPlot[Range[10], Filling → Axis]
```

Out[2]=



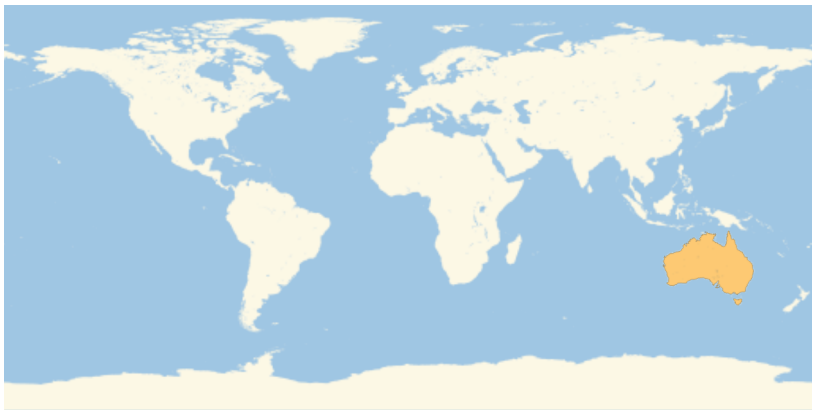
In[3]:=
```
ListPlot[Range[10], Background → Yellow]
```

Out[3]=



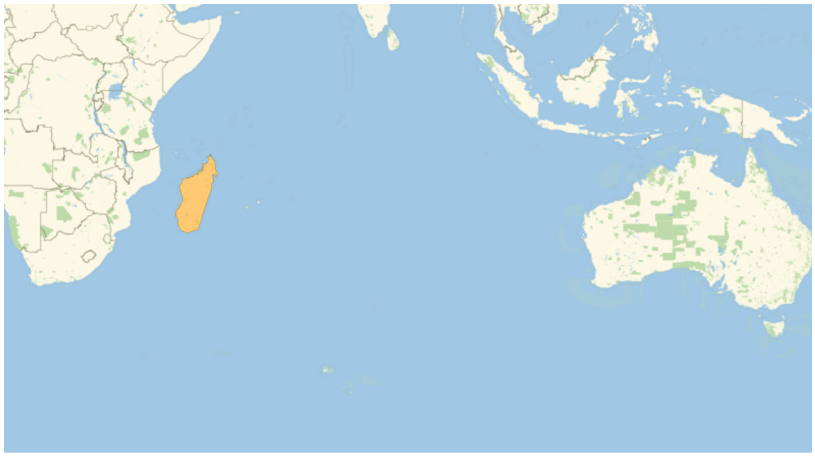
In[4]:=
```
GeoListPlot[Australia COUNTRY, GeoRange → Earth PLANET]
```

Out[4]=



In[5]:=
```
GeoListPlot[Madagascar COUNTRY, GeoRange → Indian Ocean OCEAN]
```

Out[5]=



In[6]:=
```
GeoGraphics[GeoRange → South America COUNTRIES, GeoBackground → "ReliefMap"]
```

Out[6]=

In[7]:=
```
GeoListPlot[{France COUNTRY, Finland COUNTRY, Greece COUNTRY},
 GeoLabels → True, GeoRange → Europe GEOGRAPHIC REGION]
```

Out[7]=

In[8]:=
```
GeoListPlot[ The Ivy League UNIVERSITIES , GeoLabels → Automatic]
```

Out[8]=

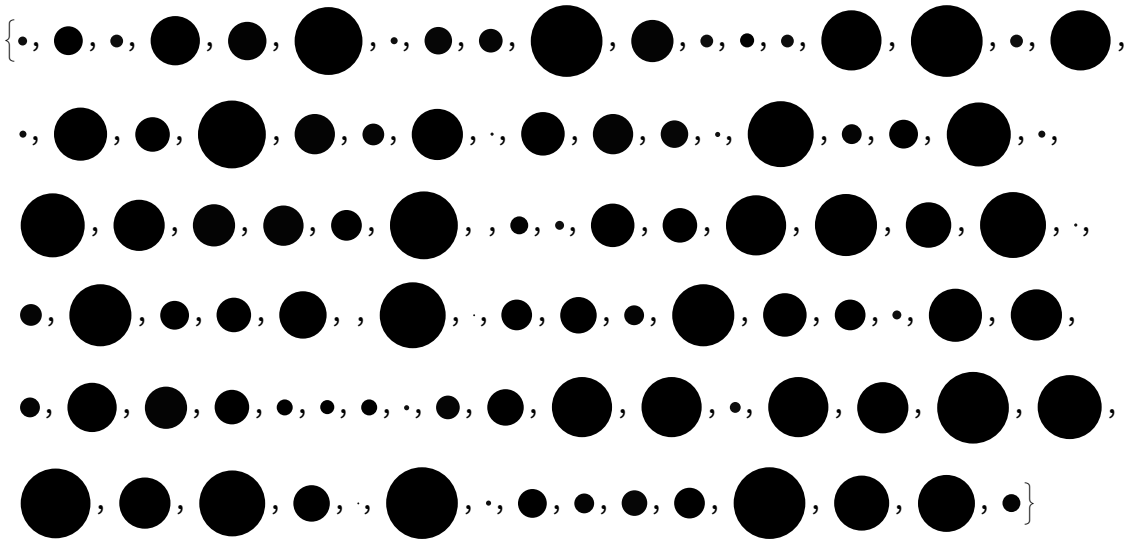
In[9]:=
```
Grid[Table[Style[n * m, White], {n, 12}, {m, 12}], Background → Black]
```

Out[9]=

1	2	3	4	5	6	7	8	9	10	11	12
2	4	6	8	10	12	14	16	18	20	22	24
3	6	9	12	15	18	21	24	27	30	33	36
4	8	12	16	20	24	28	32	36	40	44	48
5	10	15	20	25	30	35	40	45	50	55	60
6	12	18	24	30	36	42	48	54	60	66	72
7	14	21	28	35	42	49	56	63	70	77	84
8	16	24	32	40	48	56	64	72	80	88	96
9	18	27	36	45	54	63	72	81	90	99	108
10	20	30	40	50	60	70	80	90	100	110	120
11	22	33	44	55	66	77	88	99	110	121	132
12	24	36	48	60	72	84	96	108	120	132	144

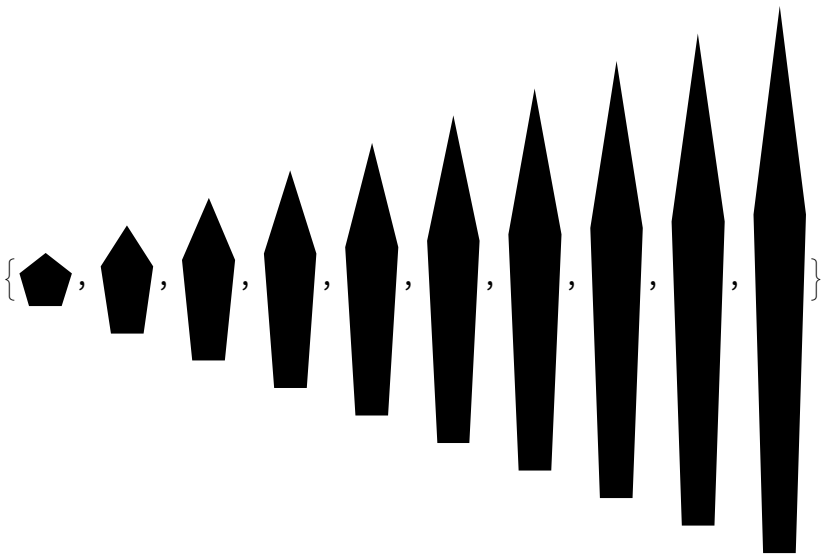
In[10]:=
```
Table[Graphics[Disk[], ImageSize → RandomInteger[40]], 100]
```

Out[10]=



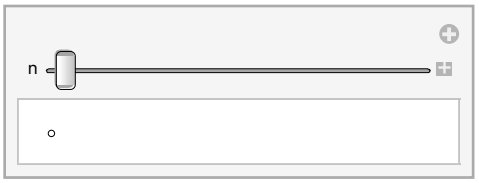
In[11]:=
```
Table[Graphics[RegularPolygon[5], ImageSize → 30, AspectRatio → n], {n, 10}]
```

Out[11]=



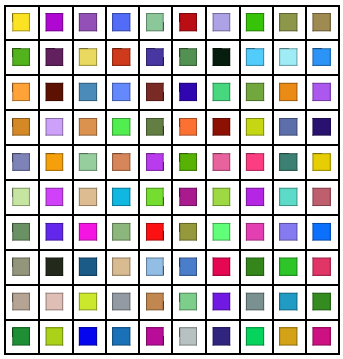
In[12]:=
```
Manipulate[Graphics[Circle[], ImageSize → n], {n, 5, 500}]
```

Out[12]=



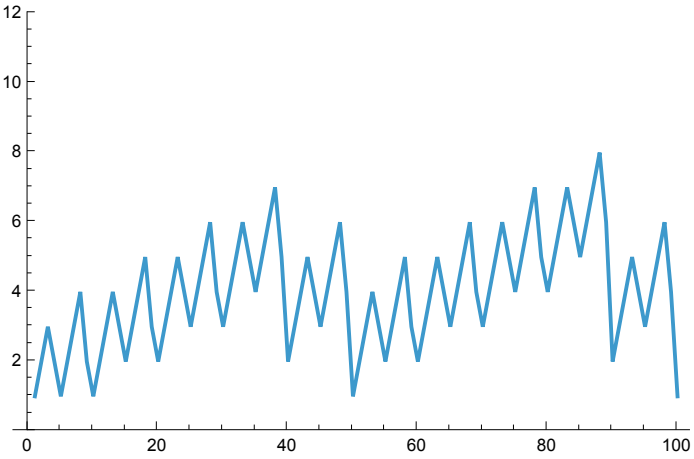
In[13]:=
```
Grid[Table[RandomColor[], 10, 10], Frame → All]
```

Out[13]=



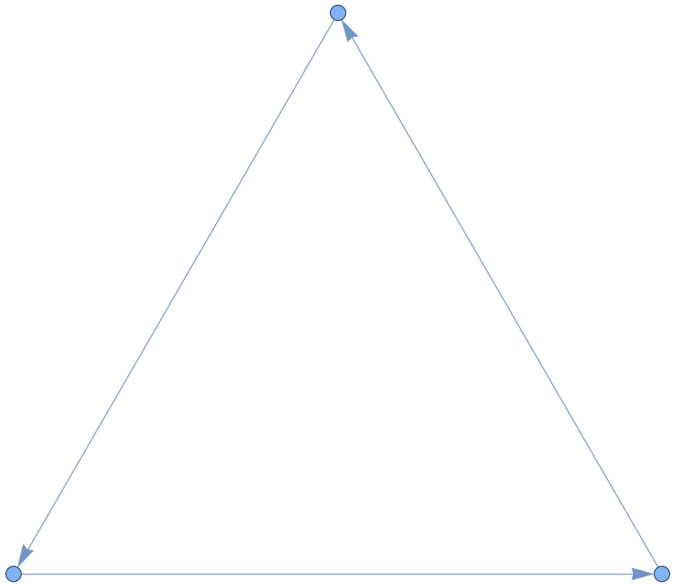
In[14]:=
```
ListLinePlot[Table[StringLength[RomanNumeral[n]], {n, 100}],
 PlotRange → Max[Table[StringLength[RomanNumeral[m]], {m, 1000}]]]
```

Out[14]=



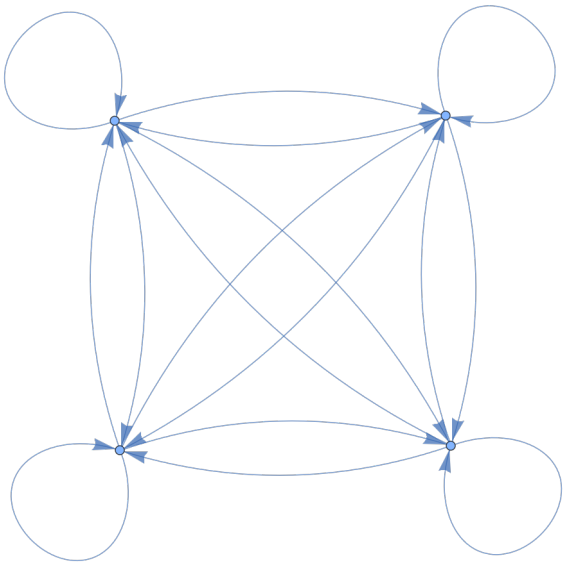
In[15]:=
```
Graph[{1 → 2, 2 → 3, 3 → 1}]
```

Out[15]=



In[16]:=
```
Graph[Flatten[Table[i → j, {i, 4}, {j, 4}]]]
```

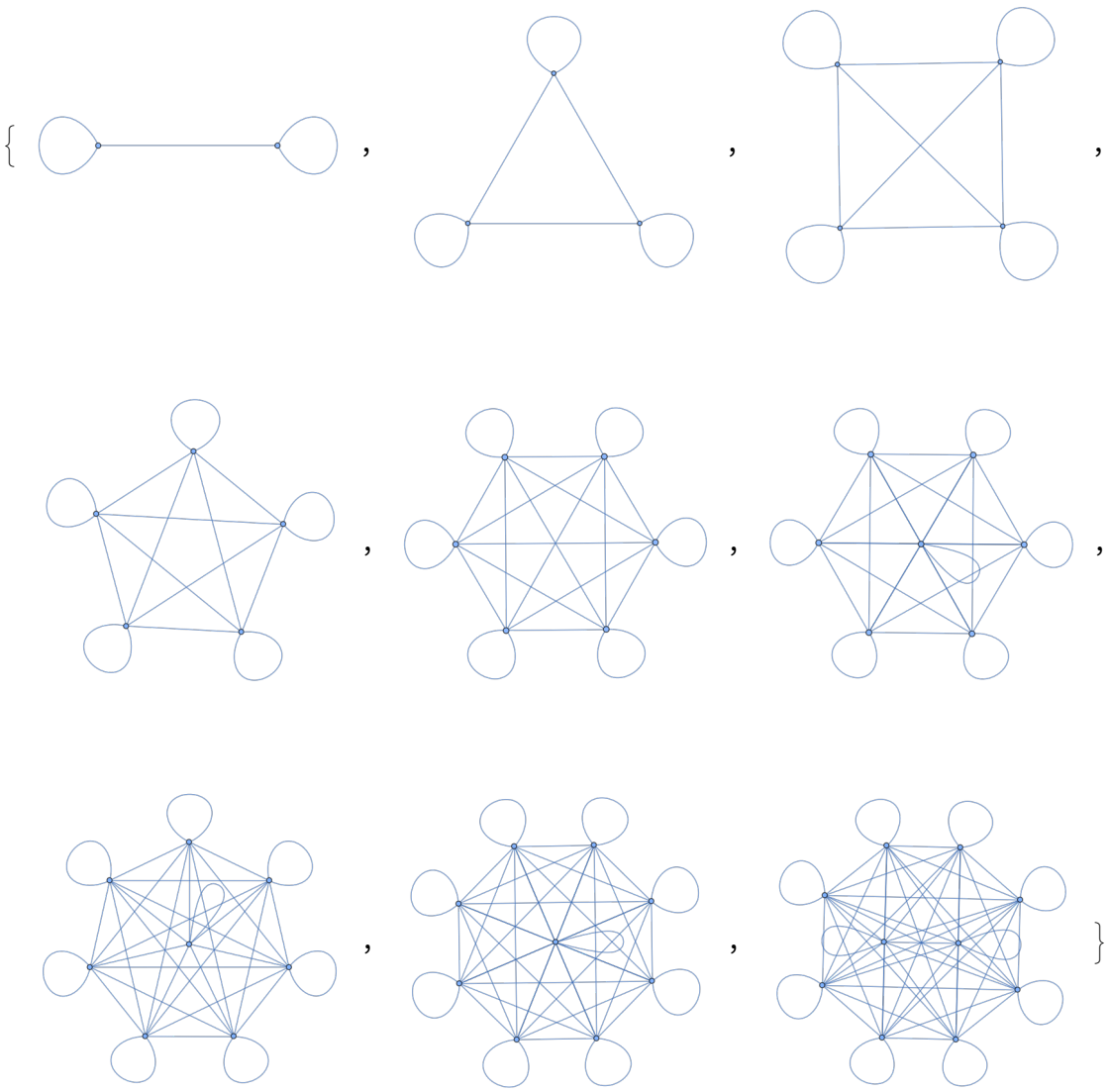
Out[16]=



I had a different and simpler interpretation of this one.

In[17]:=
```
Table[UndirectedGraph[Flatten[Table[i → j, {i, n}, {j, n}]]], {n, 2, 10}]
```

Out[17]=

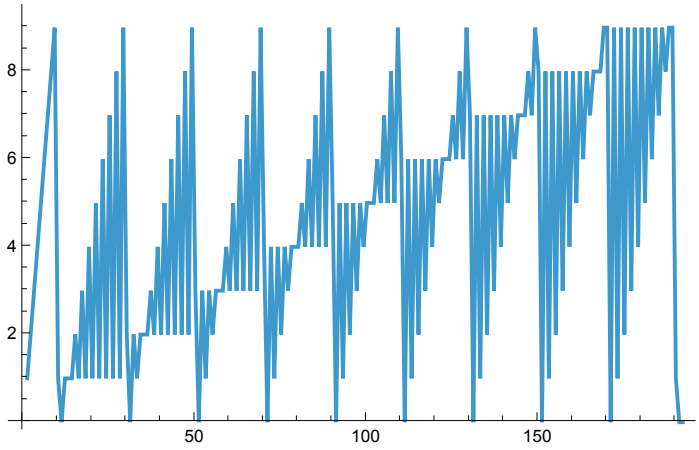


In[18]:=
```
Flatten[Table[Range[2], 3]]
```

Out[18]=
```
{1, 2, 1, 2, 1, 2}
```

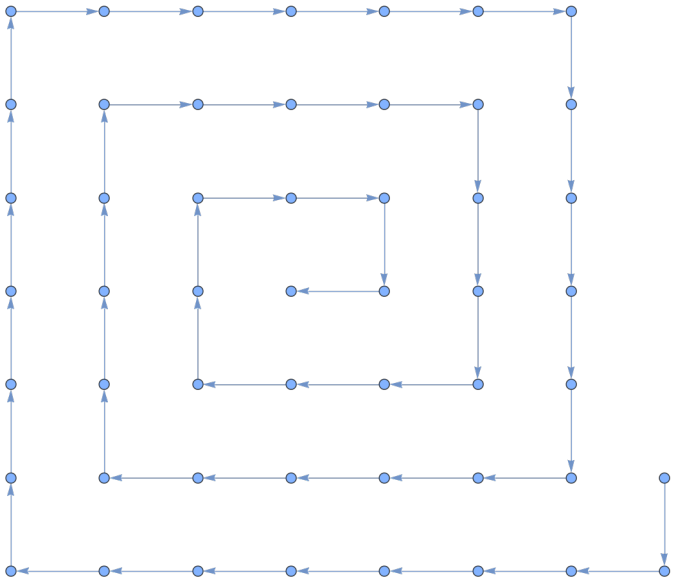
In[19]:=
```
ListLinePlot[Flatten[Table[IntegerDigits[n], {n, 100}]]]
```

Out[19]=



In[20]:=
```
Graph[Table[i → i + 1, {i, 50}]]
```

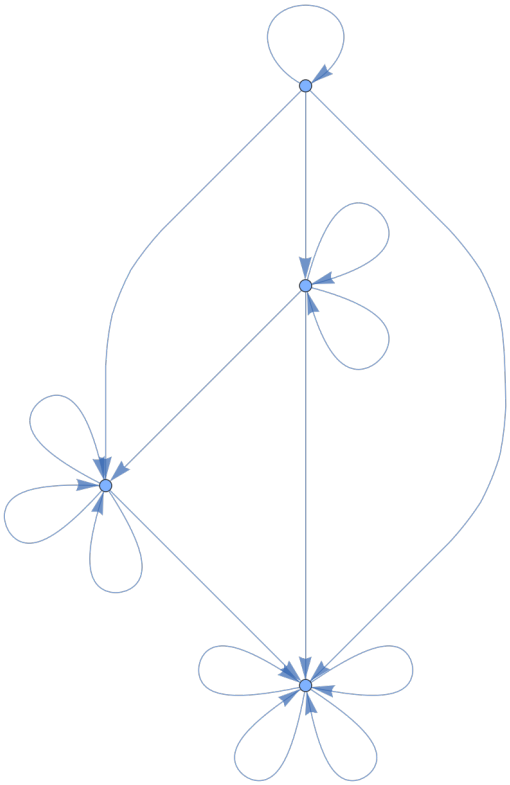
Out[20]=



You have included a 51st node.

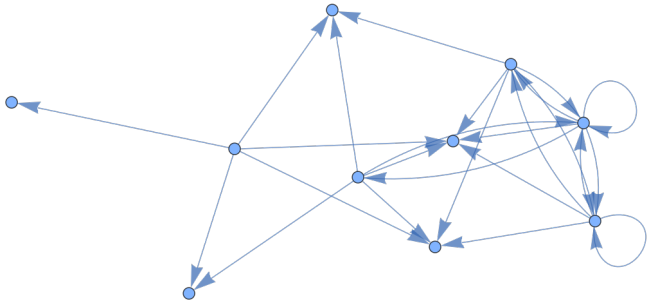
In[21]:=
```
Graph[Flatten[Table[i → Max[i, j], {i, 4}, {j, 4}]]]
```

Out[21]=



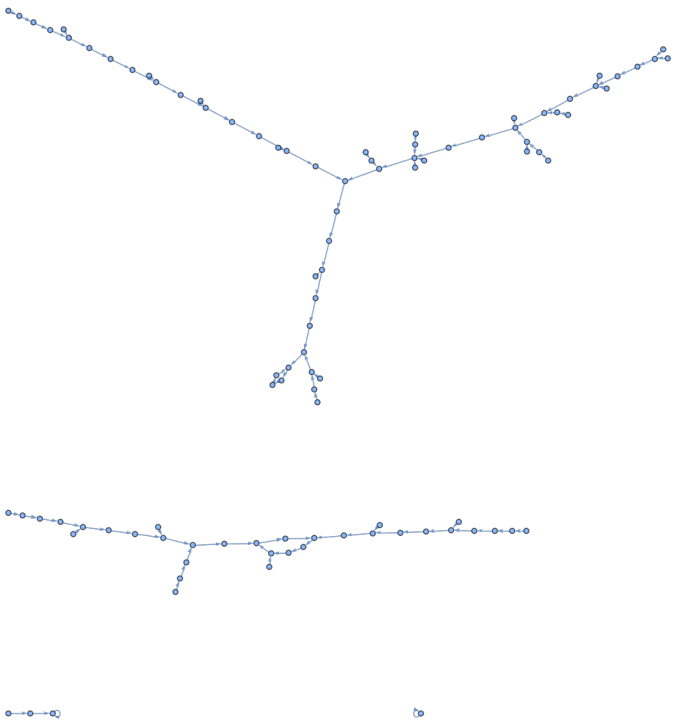
In[22]:=
```
Graph[Flatten[Table[i → j - i, {i, 5}, {j, 5}]]]
```

Out[22]=



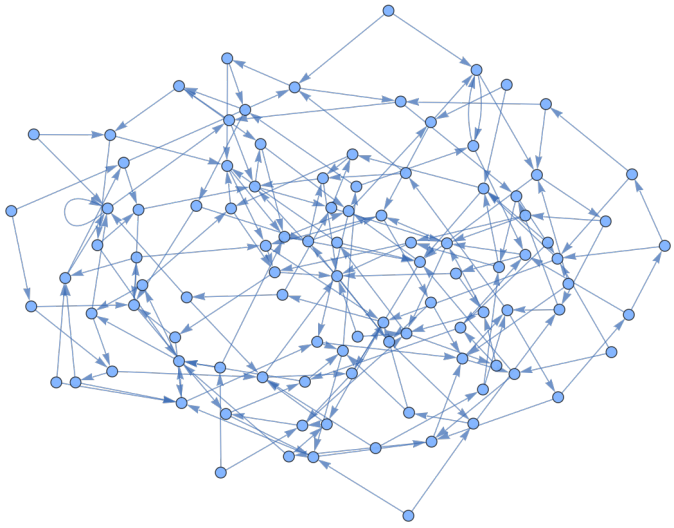
In[23]:=
```
Graph[Flatten[Table[i → RandomInteger[{1, 100}], {i, 100}]]]
```

Out[23]=



In[24]:=
```
Graph[
 Flatten[Table[{i → RandomInteger[{1, 100}], i → RandomInteger[{1, 100}]}, {i, 100}]]]
```

Out[24]=



In[25]:=
```
Grid[Table[FindShortestPath[
    Graph[{1 → 2, 2 → 3, 3 → 4, 4 → 1, 3 → 1, 2 → 2}], n, m], {n, 4}, {m, 4}]]
```

Out[25]=

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

In[26]:=
```
LanguageIdentify["ajatella"]
```

Out[26]=

Finnish

In[27]:=
```
ImageIdentify[ tiger SPECIES SPECIFICATION [ image ] ]
```

Out[27]=

tiger

In[28]:=
```
Table[ImageIdentify[Blur[ tiger SPECIES SPECIFICATION [ image ] , r]], {r, 5}]
```

Out[28]=

{ tiger , tiger , tiger , tiger , swift fox }

In[29]:=
```
Classify["Sentiment", "I'm so happy to be here"]
```

Out[29]=

Positive

In[30]:=
```
Nearest[WordList[], "happy", 10]
```

Out[30]=

{happy, haply, harpy, nappy, sappy, apply, campy, choppy, guppy, hairy}

In[31]:=
```
Nearest[Table[RandomInteger[1000], 20], 100, 3]
```

Out[31]=

{103, 169, 198}

In[32]:=
```
Nearest[Table[RandomColor[], 10], Red, 5]
```

Out[32]=

{ , , , , }

In[33]:=
```
Nearest[Table[n^2, {n, 100}], 2000]
```

Out[33]=

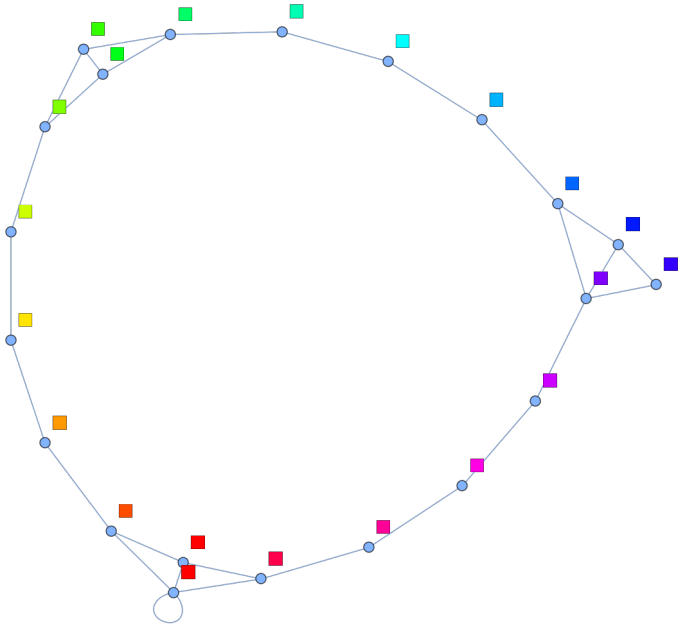
{2025}

In[34]:=
```
Nearest[Entity["Europe", "COUNTRIES"][EntityProperty["flag"]], Entity["Brazil", "COUNTRY"][EntityProperty["flag"]], 3]
```

Out[34]=

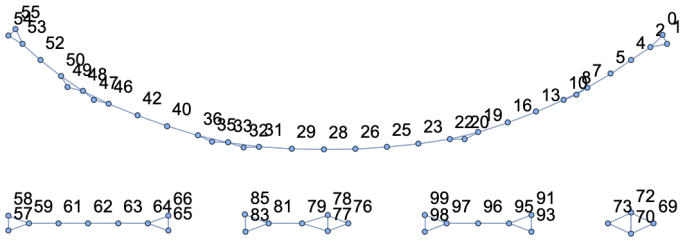
In[35]:=
```
NearestNeighborGraph[Table[Hue[h], {h, 0, 1, .05}], 2, VertexLabels → All]
```

Out[35]=



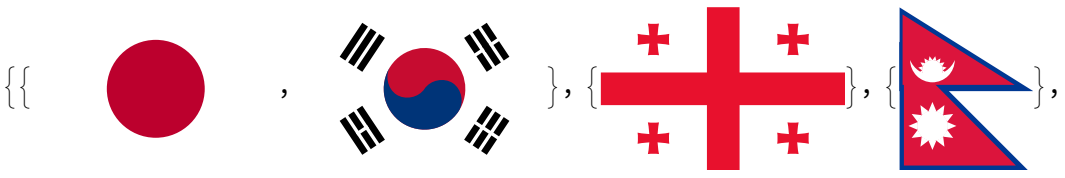
In[36]:=
```
NearestNeighborGraph[Table[RandomInteger[100], 100], 2, VertexLabels → All]
```

Out[36]=



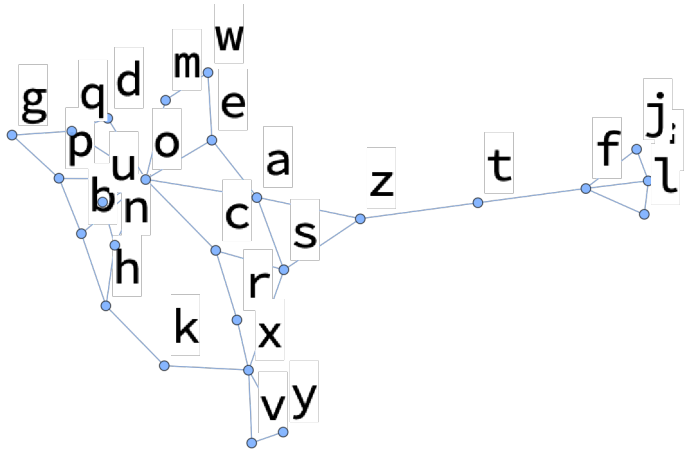
In[37]:=
```
FindClusters[Entity["Asia", "COUNTRIES"][EntityProperty["flag"]]]
```

Out[37]=



In[38]:=
```
NearestNeighborGraph[
 Table[Rasterize[Style[FromLetterNumber[n], 20]], {n, 1, 26}], 2, VertexLabels → All]
```

Out[38]=

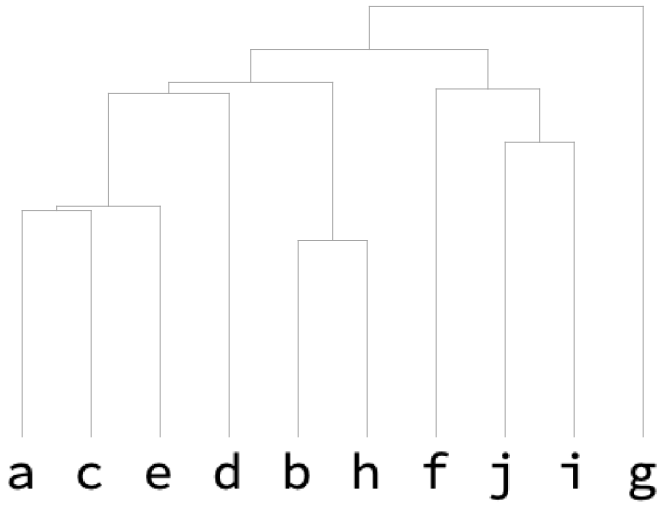


In[39]:=
```
Table[TextRecognize[EdgeDetect[Rasterize[Style["programming", n]]]], {n, 10, 20}]
```

Out[39]=
```
{orogramming, programming, programming, programming, programming,
 programming, programming, programming, programming, programming, programming}
```

In[40]:=
```
Dendrogram[Table[Rasterize[FromLetterNumber[n]], {n, 10}]]
```

Out[40]=



In[41]:=
```
FeatureSpacePlot[Table[Rasterize[ToUpperCase[FromLetterNumber[n]]], {n, 26}]]
```

Out[41]=