Brian — PS 9 — 2025-02-21 — Solution

EIWL3 Sections 23, 24, and 25

Exercises from *EIWL3* Section 23

In[57]:=
```
(* 23.1 *) N[Sqrt[2], 500]
```

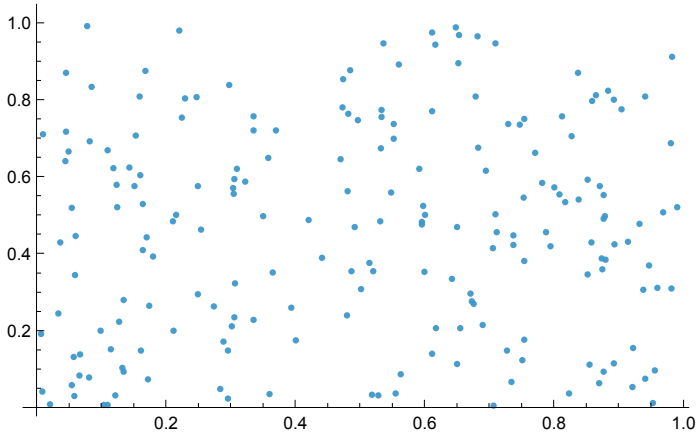
Out[57]=
```
1.41421356237309504880168872420969807856967187537694807317667973799073247846210703885038753432764157273501384623091229702492483605585073721264412149709993583141322266592750559275579995050115278206057147010955997160597027453459686201472851741864088919860955232923048430871432145083976260362799525140798968725339654633180882964062061525835239505474575028775996172983557522033753185701135437460340849884716038689997069900481503054402779031645424782306849293691862158057846311159666871301301561856898723721.
```

In[58]:=
```
(* 23.2 *) RandomReal[1, 10]
```

Out[58]=
```
{0.961149, 0.0834804, 0.530357, 0.515936, 0.423071,
 0.00491769, 0.0491434, 0.133422, 0.137943, 0.381342}
```

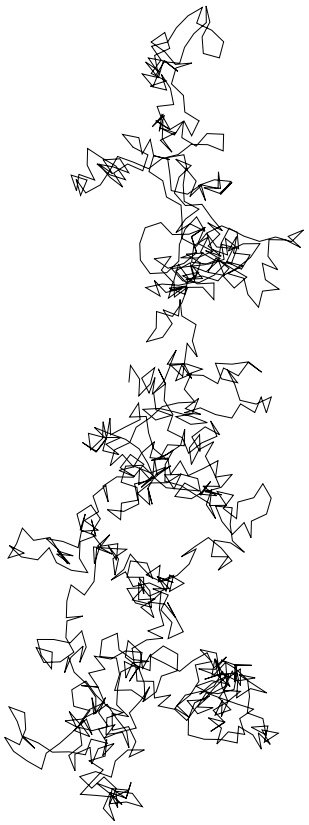
In[59]:=
```
(* 23.3 *) ListPlot[RandomReal[1, {200, 2}]]
```

Out[59]=



In[60]:=
```
(* 23.4 *) Graphics[Line[AnglePath[RandomReal[2 Pi, 1000]]]]
```

Out[60]=



In[61]:=
```
(* 23.5 *) Table[Mod[n^2, 10], {n, 0, 30}]
```

Out[61]=
```
{0, 1, 4, 9, 6, 5, 6, 9, 4, 1, 0, 1, 4, 9, 6, 5, 6, 9, 4, 1, 0, 1, 4, 9, 6, 5, 6, 9, 4, 1, 0}
```

In[62]:=
```
(* 23.6 *) Table[Mod[n^n, 10], {n, 100}]
```

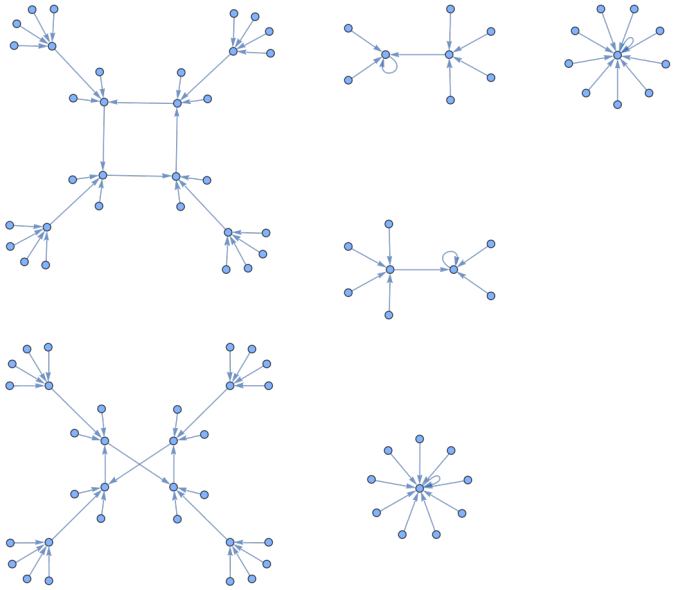
Out[62]=
```
{1, 4, 7, 6, 5, 6, 3, 6, 9, 0, 1, 6, 3, 6, 5, 6, 7, 4, 9, 0, 1, 4, 7, 6,
 5, 6, 3, 6, 9, 0, 1, 6, 3, 6, 5, 6, 7, 4, 9, 0, 1, 4, 7, 6, 5, 6, 3, 6, 9,
 0, 1, 6, 3, 6, 5, 6, 7, 4, 9, 0, 1, 4, 7, 6, 5, 6, 3, 6, 9, 0, 1, 6, 3, 6,
 5, 6, 7, 4, 9, 0, 1, 4, 7, 6, 5, 6, 3, 6, 9, 0, 1, 6, 3, 6, 5, 6, 7, 4, 9, 0}
```

In[63]:=
```
(* 23.7 *) Round[Table[Pi^i, {i, 10}]]
```

Out[63]=
```
{3, 10, 31, 97, 306, 961, 3020, 9489, 29 809, 93 648}
```

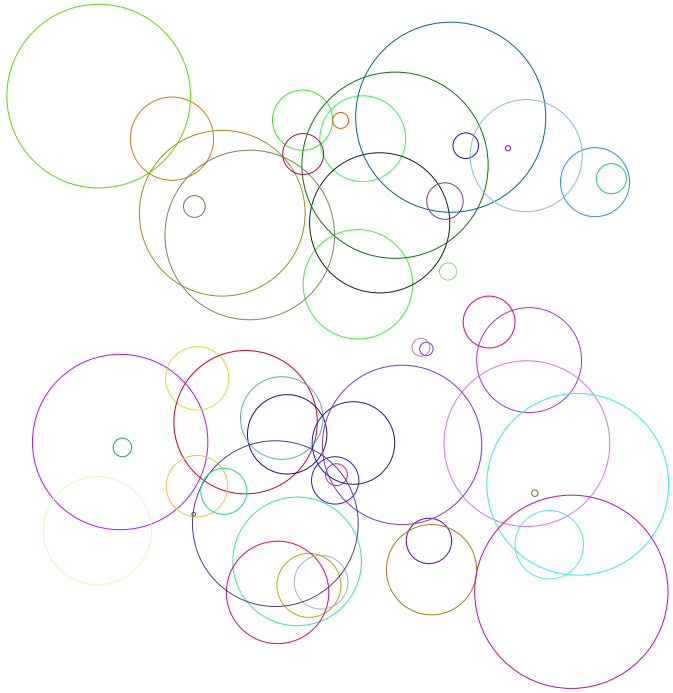
In[64]:=
```
(* 23.8 *) Graph[Table[n → Mod[n^2, 100], {n, 0, 99}]]
```

Out[64]=



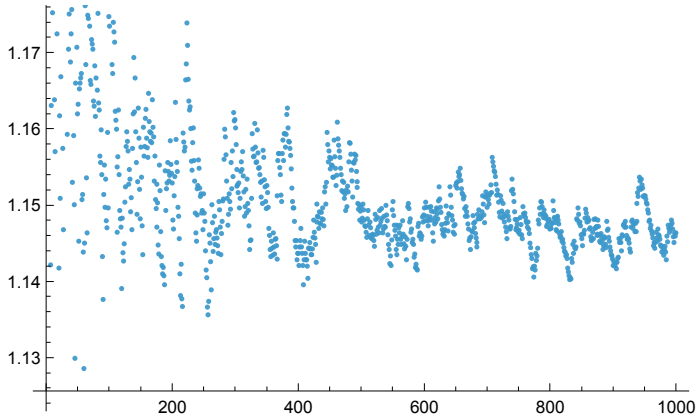
In[65]:=
```
(* 23.9 *) Graphics[Table[
  Style[Circle[{RandomReal[10], RandomReal[10]}, RandomReal[2]], RandomColor[]],
  50]]
(* This is an expression that is just complicated enough that
 I decided to use indenting to help me write it out correctly. *)
```

Out[65]=



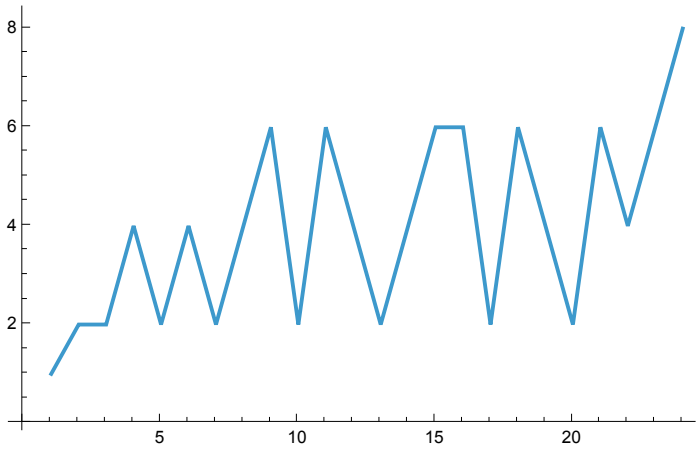
In[66]:=
```
(* 23.10 *) ListPlot[Table[Prime[n]/(n Log[n]), {n, 2, 1000}]]
```

Out[66]=



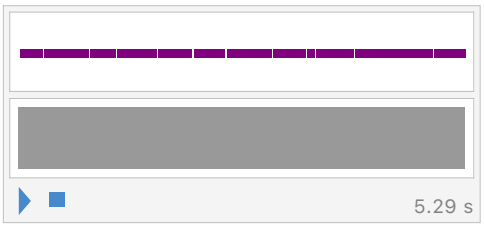
In[67]:=
```
(* 23.11 *) ListLinePlot[Table[Prime[n] - Prime[n - 1], {n, 2, 25}]]
(* I am not sure how we were supposed to know that the 25th prime was *)
(* the last one less than 100. I just did trial and error to figure that out. *)
```

Out[67]=



In[68]:=
```
(* 23.12 *) Sound[Table[SoundNote[0, RandomReal[0.5]], 20]]
```

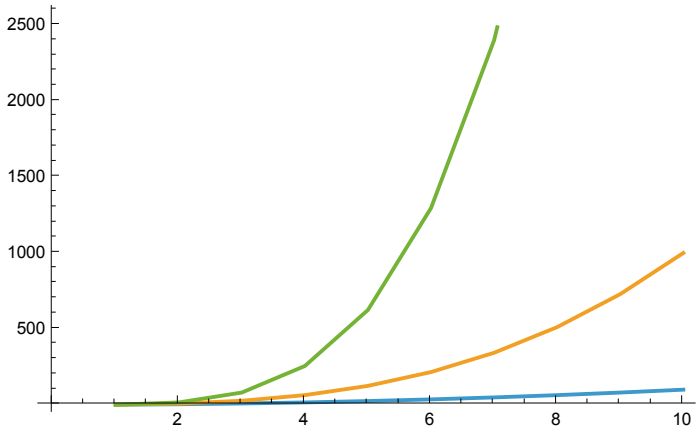
Out[68]=



Exercises from *EIWL3* Section 24

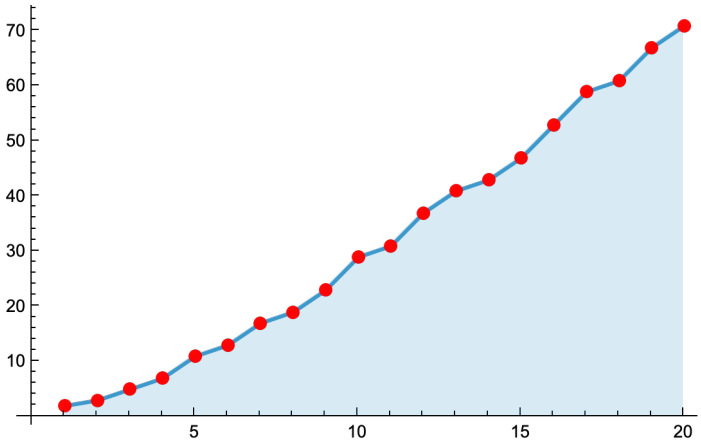
In[69]:=
```
(* 24.1 *) ListLinePlot[{Range[10]^2, Range[10]^3, Range[10]^4}]
```

Out[69]=



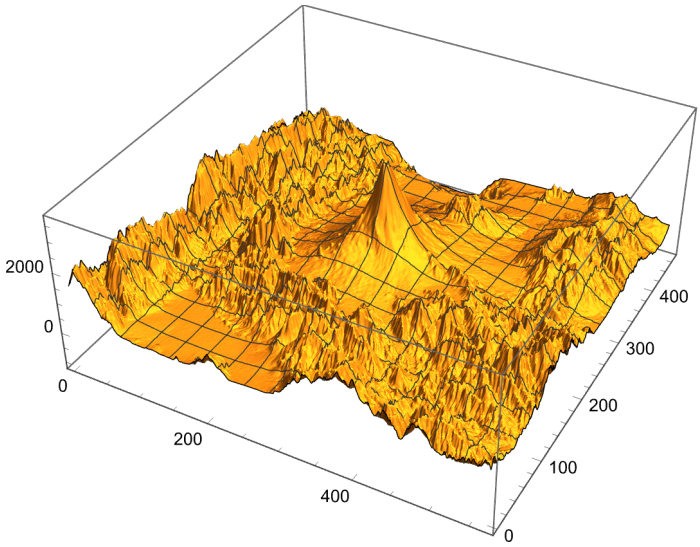
In[70]:=
```
(* 24.2 *) ListLinePlot[Table[Prime[i], {i, 20}],
 Filling → Axis, Mesh → All, MeshStyle → Red]
```

Out[70]=



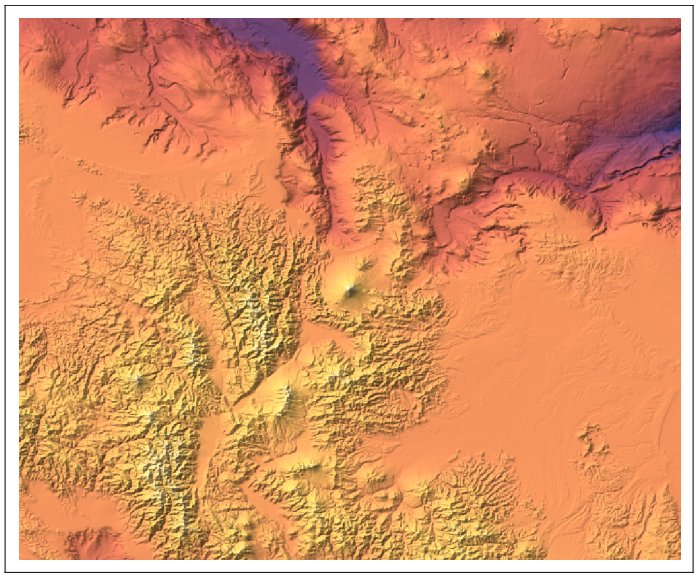
In[71]:=
```
(* 24.3 *)
ListPlot3D[GeoElevationData[GeoDisk[ Mount Fuji MOUNTAIN , 20 mi ]], PlotRange → All]
```

Out[71]=



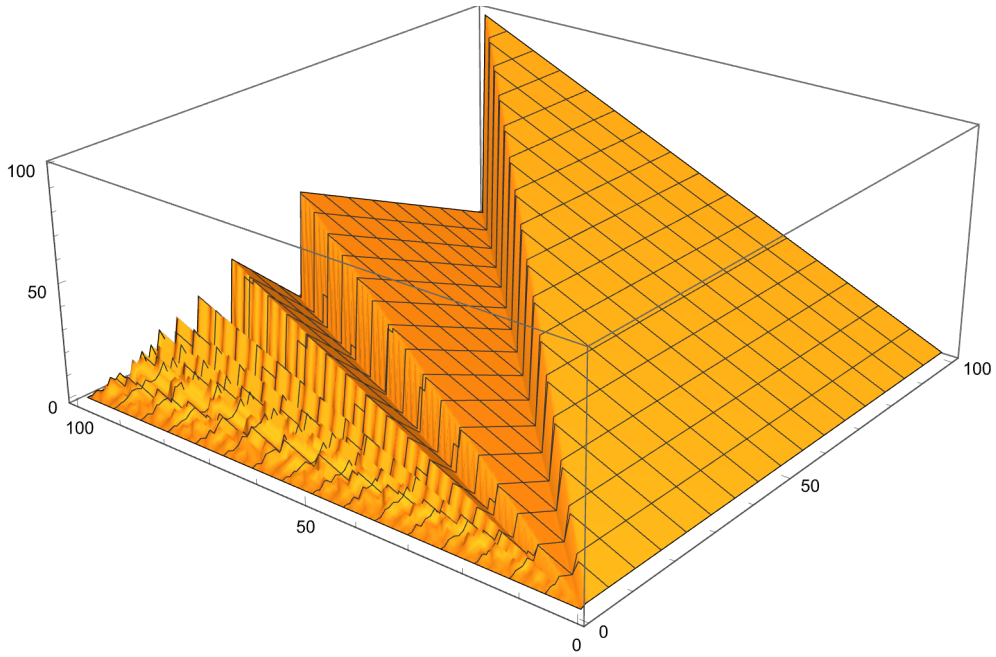
In[72]:=
```
(* 24.4 *) ReliefPlot[GeoElevationData[GeoDisk[ Mount Fuji MOUNTAIN , 100 mi ]]]
```

Out[72]=



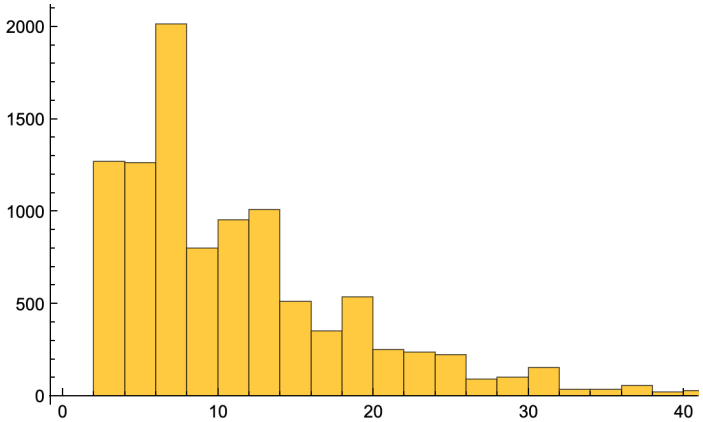
In[73]:=
```
(* 24.5 *) ListPlot3D[Table[Mod[i, j], {i, 100}, {j, 100}]]
```

Out[73]=



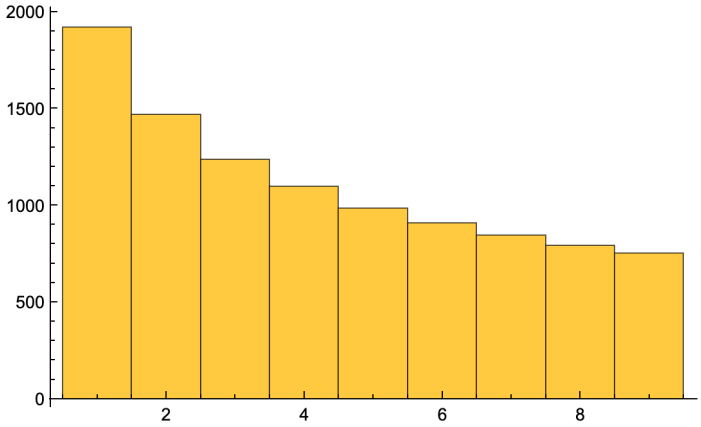
In[74]:=
```
(* 24.6 *) Histogram[Table[Prime[j + 1] - Prime[j], {j, 9999}]]
```

Out[74]=



In[75]:=
```
(* 24.7 *) Histogram[Table[First[IntegerDigits[j^2]], {j, 10 000}]]
```

Out[75]=

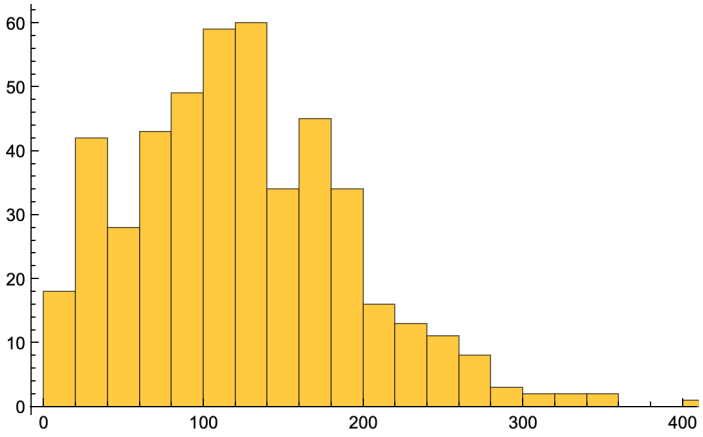


In[76]:=
```
(* 24.8 *) Histogram[Table[StringLength[RomanNumeral[i]], {i, 1000}]]
```

Out[76]=

In[77]:=
```
(* 24.9 *) Histogram[StringLength[TextSentences[WikipediaData["Computers"]]]]
```

Out[77]=

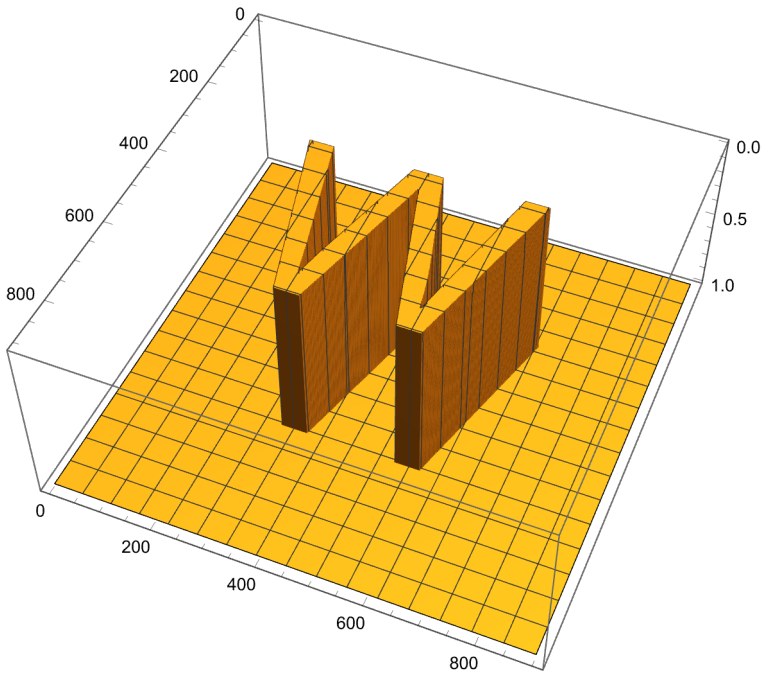


In[78]:=
```
(* 24.10 *) Table[Table[Total[RandomReal[100.0, n]], 2], {n, 1, 5}]
```

Out[78]=
```
{{40.8522, 14.8525}, {118.795, 149.493},
 {90.923, 160.367}, {187.113, 244.849}, {202.133, 157.38}}
```

In[79]:=
```
(* 24.11 *) ListPlot3D[ImageData[Binarize[Graphics[Style[Text["W"], 200]]]]]
```

Out[79]=



Exercises from *EIWL3* Section 25

In[80]:=
```
(* 25.1 *) f /@ Range[5]
```

Out[80]=
```
{f[1], f[2], f[3], f[4], f[5]}
```

In[81]:=
```
(* This is indeed the same result as *) Table[f[n], {n, 5}]
(* We now eschew the bloated code we first learned. *)
(* This is going to eliminate a lot of our uses of Table[]. *)
```

Out[81]=
```
{f[1], f[2], f[3], f[4], f[5]}
```

In[82]:=
```
(* 25.2 *) f /@ g /@ Range[10]
```

Out[82]=
```
{f[g[1]], f[g[2]], f[g[3]], f[g[4]],
 f[g[5]], f[g[6]], f[g[7]], f[g[8]], f[g[9]], f[g[10]]}
```

In[83]:=
```
(* 25.3 *)x // d // c // b // a
```

Out[83]=
```
a[b[c[d[x]]]]
```

In[84]:=
```
(* This is indeed the same result as *) a[b[c[d[x]]]]
```

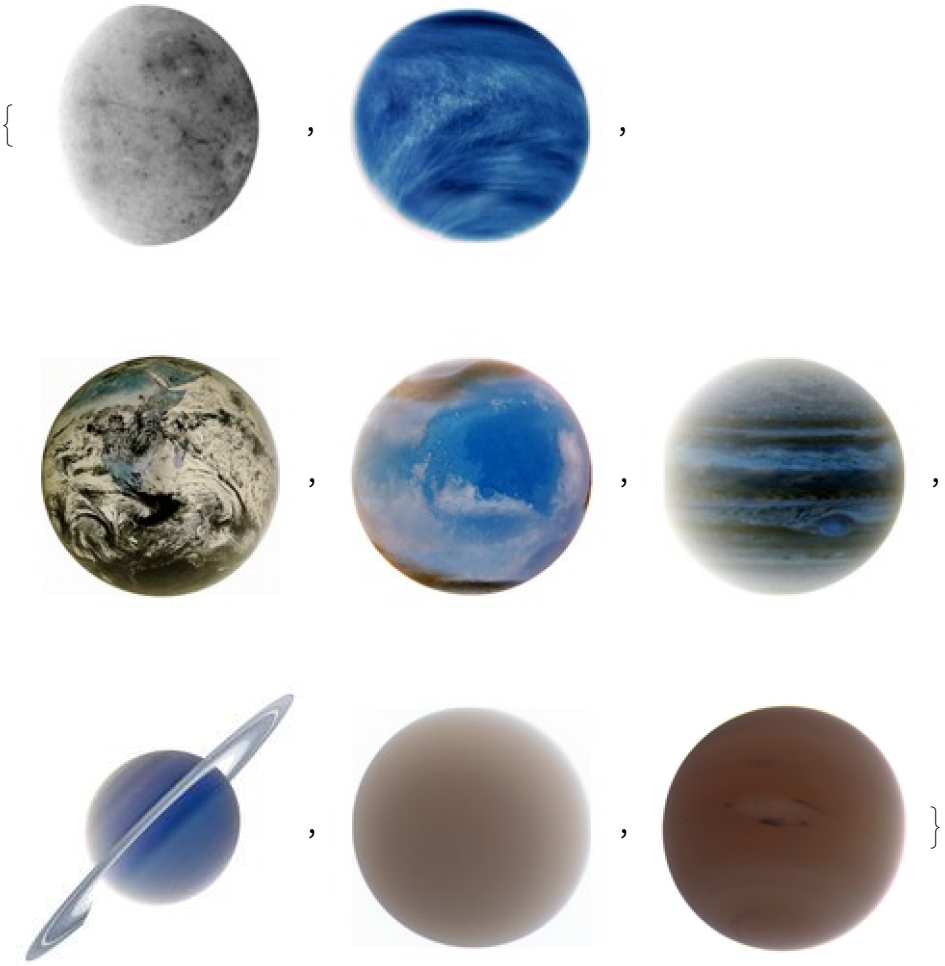
Out[84]=
```
a[b[c[d[x]]]]
```

In[85]:=
```
(* 25.4 *) Framed /@ Alphabet[]
```

Out[85]=
```
{a, b, c, d, e, f, g, h, i, j, k, l,
 m, n, o, p, q, r, s, t, u, v, w, x, y, z}
```

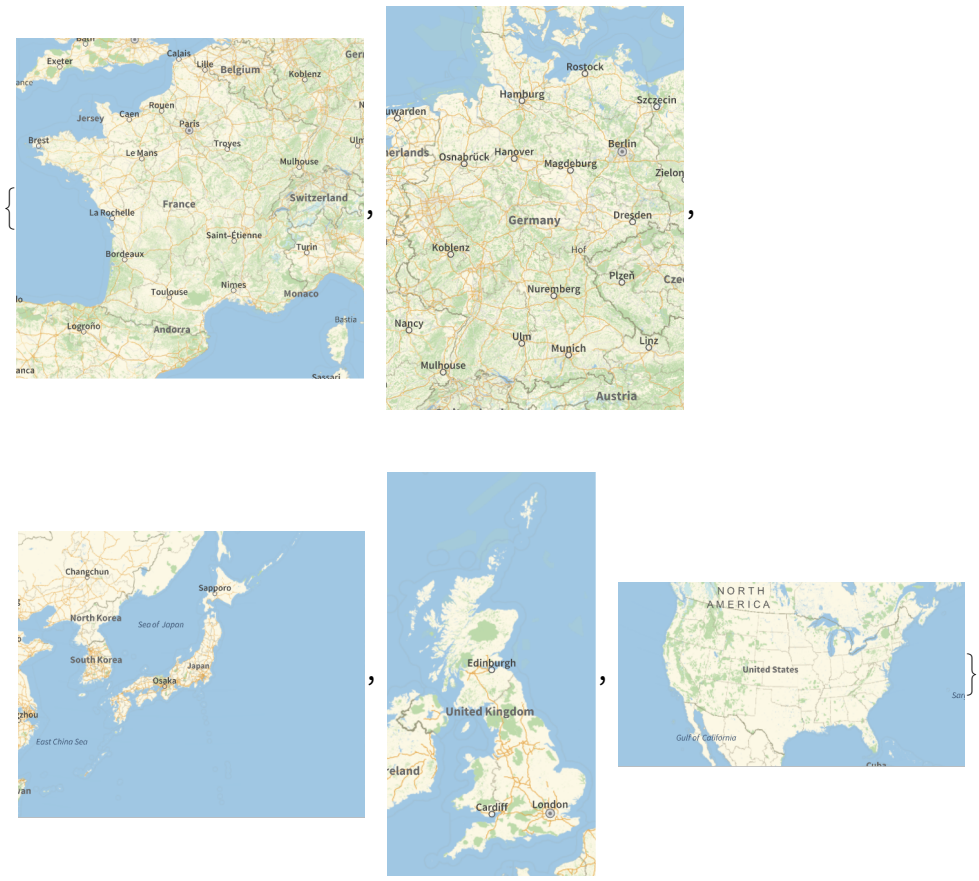
In[86]:=
```
(* 25.5 *)ColorNegate /@ planets PLANETS [ image ]
```

Out[86]=



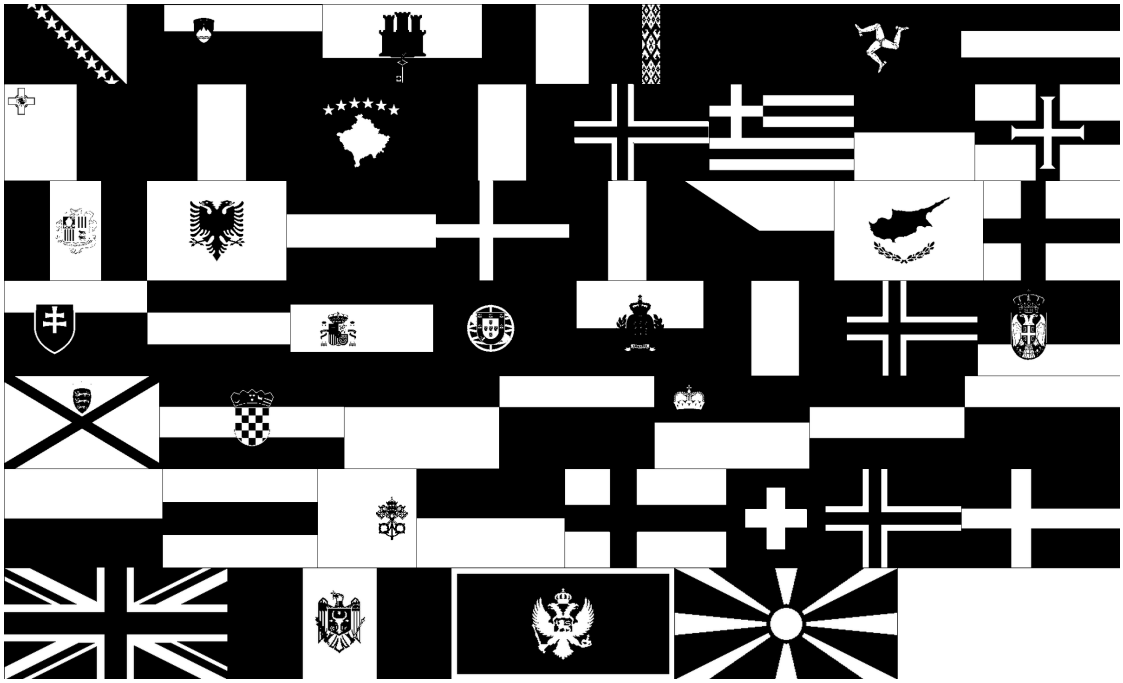
In[87]:=
```
(* 25.6 *) GeoGraphics /@ EntityList[ Group of 5 COUNTRIES ]
(* We learned about EntityList[] back in Section 16, *)
(* but I will be frank and admit that I could not *)
(* remember that I first needed to call EntityList[], *)
(* and even more frankly, I'll admit that I am still *)
(* unclear when EntityList[] and EntityValue are needed. *)
```

Out[87]=



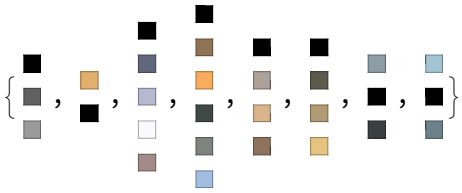
In[88]:=
```
(* 25.7 *) Binarize /@ Europe COUNTRIES [ flag ] // ImageCollage
```

Out[88]=



In[89]:=
```
(* 25.8 *) Column /@ DominantColors /@ planets PLANETS [ image ]
```

Out[89]=



In[90]:=
```
(* 25.9 *) Total[LetterNumber["Wolfram"]]
```

Out[90]=

88