Brian — PS 14 — 2025-03-28 — Solution

EIWL3 Sections 35 and 36

Exercises from *EIWL3* Section 35

```
In[•]:= (* 35.1 *) Interpreter["Location"]["Eiffel Tower"]
```

```
Out[•]=
GeoPosition[{48.8583, 2.29444}]
```

```
(* 35.2 *) Interpreter["University"]["U of T"]
```

```
Out[•]=
University of Toronto
```

```
In[•]:= (* 35.3 *) Interpreter["Chemical"][{"C2H4", "C2H6", "C3H8"}]
```

```
Out[•]=
{ethylene, ethane, propane}
```

```
In[•]:= (* 35.4 *) Interpreter["Date"]["20140108"]
```

```
Out[•]=
Wed 8 Jan 2014
```

```
(* 35.5 *) (* I was able to get this far: *)
Interpreter["University"][
 Table[StringJoin["U of ", letter], {letter, Capitalize[Alphabet[]]}]]
(* See below *)
```

```
Out[•]=
{Failure[⚠ Message: No university interpretation found. Try again.
         Tag: InterpretationFailure],
 University of Birjand, University of California-Los Angeles,
 Failure[⚠ Message: No university interpretation found. Try again.
           Tag: InterpretationFailure],
 The University of Edinburgh, Failure[⚠ Message: No university interpretation found. Try again.
                                       Tag: InterpretationFailure],
 University of Georgia, University of Houston, University of Illinois at Urbana-Champaign,
 Failure[⚠ Message: No university interpretation found. Try again.
           Tag: InterpretationFailure],
```

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

University of Lethbridge , University of Minnesota-Twin Cities ,

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

University of Phoenix-Online Campus ,

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

University of Regina , University of Saskatchewan , University of Toronto ,

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure],

Failure[⚠ Message: No university interpretation found. Try again.
Tag: InterpretationFailure] }

In[ ]:=
```
(* Then I had to look up how to proceed. The trick is to wrap *)
(* the results in Cases[ ... , _Entity]. This eliminates all *)
(* the failures. *)
Cases[Interpreter["University"][
  Table[StringJoin["U of ", letter], {letter, Capitalize[Alphabet[]]}]], _Entity]
```

Out[ ]=

{ University of Birjand , University of California-Los Angeles , The University of Edinburgh ,
University of Georgia , University of Houston , University of Illinois at Urbana-Champaign ,
University of Lethbridge , University of Minnesota-Twin Cities , University of Phoenix-Online Campus ,
University of Regina , University of Saskatchewan , University of Toronto }

In[•]:=
```
(* 35.6 *) CommonName /@ Cases[Interpreter["Movie"][
  CommonName /@ all US states with District of Columbia ADMINISTRATIVE DIVISIONS [capital city]], _Entity]
```

Out[•]=
```
{Phoenix, Honolulu, Topeka, Annapolis, Lincoln, Santa Fe, Expedition: Bismarck,
 Columbus, Providence, Nashville, Olympia, Madison, Cheyenne}
```

In[•]:=
```
(* 35.7 *) Cases[
 Interpreter["Movie"][StringJoin /@ Permutations[{"a", "i", "l", "m"}]], _Entity]
```

Out[•]=
```
{Liam, Mai, Apples, Mai}
```

In[•]:=
```
(* 35.8 *) TextCases[WikipediaData["gunpowder"], "Country"]
```

Out[•]=
```
{China, China, China, Chinese, Chinese, Chinese, Chinese, China, China,
 Japan, Mongols, Syrian, Chinese, Chinese, Chinese, Chinese, China, Syria,
 Egypt, Persia, Chinese, Persian, Chinese, Mongols, Mongols, Chinese,
 Turkish, Chinese, Turkish, Chinese, Turkish, Greek, Mongols, Mongols,
 Mongols, Mongols, China, Mongols, Japan, Italy, Germany, Italy, France,
 France, France, France, Ireland, German, Great Britain, India, India,
 India, Mongols, Mongol, India, Mongol, Mongol, India, India, Portuguese,
 Bengal, French, Portuguese, French, India, Chinese, Mongol, Chinese, Mongol,
 Portuguese, Portuguese, Portuguese, Spaniards, German, China, China, China,
 China, Chinese, China, China, Peru, Chile, Peru, French, British, Germany,
 French, U, S, United Kingdom, United States, China, United States, German,
 United Kingdom, US, Great Britain, American, Great Britain, United States,
 Indian, Chilean, Great Britain, Germany, United States, Australia, UK}
```

In[•]:=
```
(* 35.9 *) TextCases["She sells seashells by the sea shore.", "Noun"]
```

Out[•]=
```
{seashells, sea, shore}
```

In[•]:=
```
StringTake[WikipediaData["computers"], 1000]
```

Out[•]=

A computer is a machine that can be programmed to automatically carry out sequences of arithmetic or logical operations (computation). Modern digital electronic computers can perform generic sets of operations known as programs. These programs enable computers to perform a wide range of tasks. The term computer system may refer to a nominally complete computer that includes the hardware, operating system, software, and peripheral equipment needed and used for full operation; or to a group of computers that are linked and function together, such as a computer network or computer cluster. It is sometimes named general purpose computer to distinguish it from a computer appliance.

A broad range of industrial and consumer products use computers as control systems, including simple special-purpose devices like microwave ovens and remote controls, and factory devices like industrial robots. Computers are at the core of general-purpose devices such as personal computers and mobile devices such

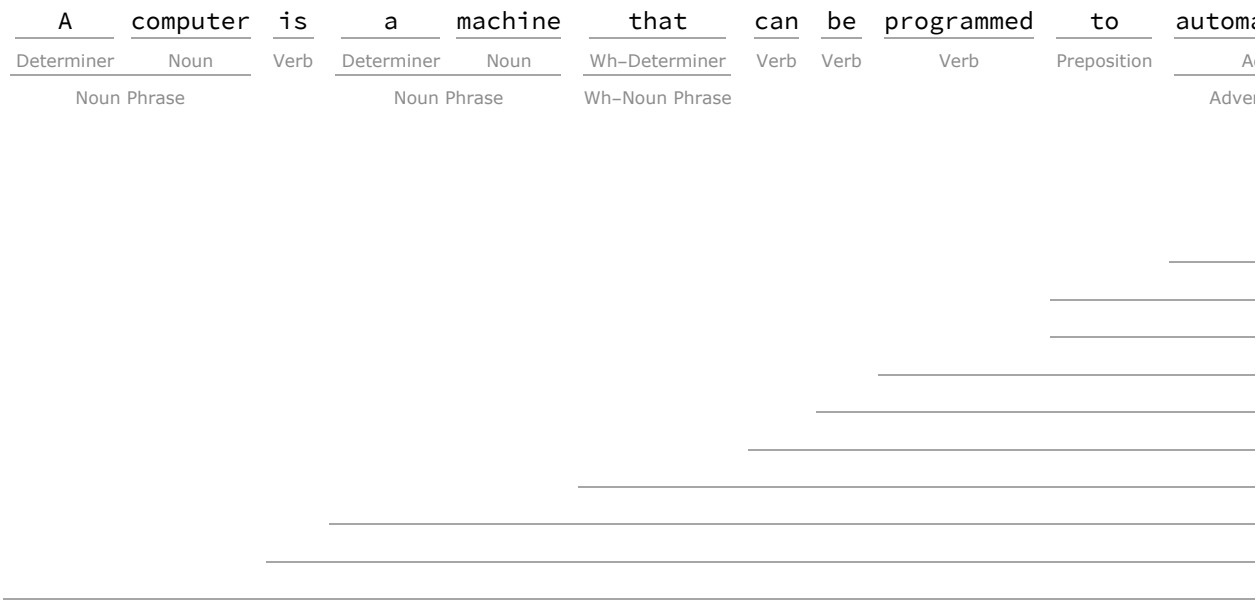
In[•]:=
```
(* 35.10 *) Length[TextCases[StringTake[WikipediaData["computers"], 1000], #]] & /@
 {"Noun", "Verb", "Adjective"}
```

Out[•]=

{51, 25, 21}

In[•]:=
```
(* 35.11 *) TextStructure[TextSentences[WikipediaData["computers"]][[1]]]
```

Out[•]=



```
(* 35.12 *)Keys[Take[Reverse[Sort[
    Counts[TextCases[ExampleData[{"Text", "AliceInWonderland"}], "Noun"]]]], 10]]
```

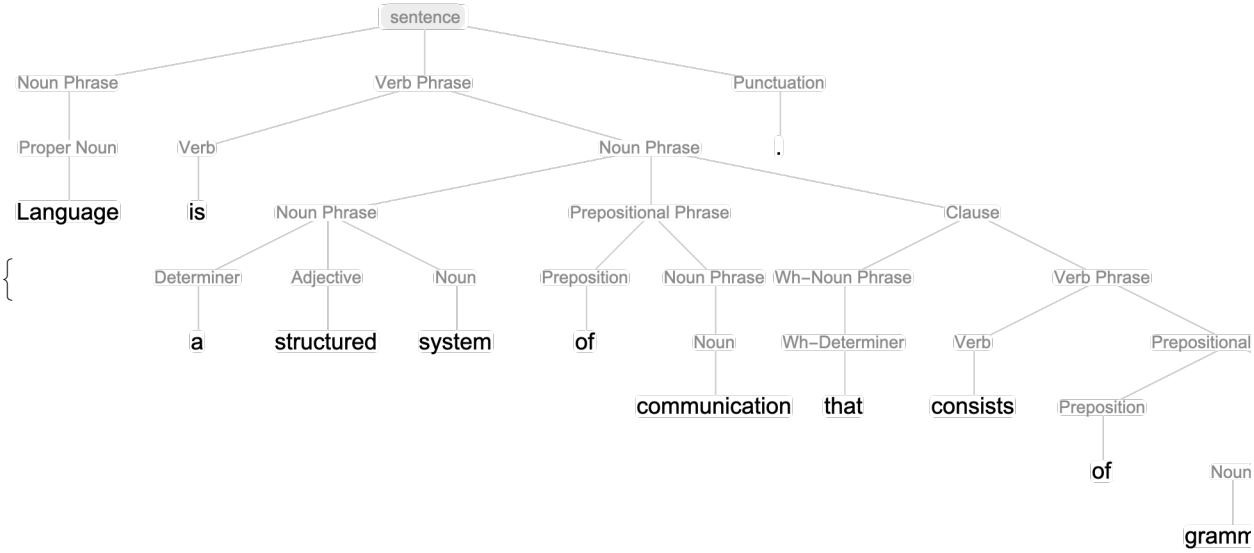
Out[•]=

```
{Rabbit, door, voice, time, Mouse, way, moment, thing, head, garden}
```

In[•]:=

```
(* 35.13 *)
TextStructure[TextSentences[WikipediaData["language"]][[1]], "ConstituentGraphs"]
```

Out[•]=



In[•]:=

```
(* 35.14 *) Length[Flatten[TextCases[WordList[], #]]] & /@
 {"Noun", "Verb", "Adjective", "Adverb"}
```

Out[•]=

```
{22 728, 5894, 7146, 2824}
```

In[•]:=

```
(* 35.15 *) WordTranslation[#, "French"] & /@ IntegerName /@ Range[2, 10]
```

Out[•]=

```
{{deux}, {trois}, {quatre}, {cinq}, {six}, {sept}, {huit}, {neuf}, {dix}}
```

Exercises from *EIWL3* Section 36

```
(* 36.1 *) CloudPublish[Delayed[Style[RandomInteger[1000], 100]]]
```

Out[14]=

```
CloudObject[https://www.wolframcloud.com/obj/7694b6b1-28d1-4c48-b21b-7b0ce3421820]
```

In[17]:=

```
(* 36.2 *)CloudPublish[FormFunction[{"number" → "Number"}, #number^2 &]]
```

Out[17]=

```
CloudObject[https://www.wolframcloud.com/obj/4962a88d-1a30-4f0c-83eb-f3d84523ba65]
```

```
(* 36.3 *)CloudPublish[FormFunction[{"x" → "Number", "y" → "Number"}, #x #y &]]
```

Out[18]=

```
CloudObject[https://www.wolframcloud.com/obj/719635db-16e1-478d-8290-5fb7d1a7a273]
```

```
(* 36.4 *) CloudPublish[
 FormFunction[{"topic" → "String"}, WordCloud[TextWords[WikipediaData[#topic]]] &]]
```

Out[20]=

CloudObject[https://www.wolframcloud.com/obj/f1fa8a38-514e-488d-a71a-f38f3338afd9]

```
(* 36.5 *) (* I don't know what Wolfram meant by "repeatedly. *)CloudPublish[
 FormFunction[{"string" → String}, Style[StringReverse[#string], 100] &]]
```

Out[26]=

CloudObject[https://www.wolframcloud.com/obj/08a80f38-8d0c-40f8-a869-a06d06b281d1]

```
(* 36.6 *)CloudPublish[FormFunction[{"n" → Integer},
  Graphics[Style[RegularPolygon[#n], RandomColor[]]] &]]
```

Out[37]=

CloudObject[https://www.wolframcloud.com/obj/f92920c8-a2f3-4dff-9371-5c6f9400fc53]

```
(* 36.7 *)CloudPublish[FormFunction[{"location" → Location, "count" → Integer},
  GeoListPlot[GeoNearest["Volcano", #location, #count]] &]]
```

Out[46]=

CloudObject[https://www.wolframcloud.com/obj/cce88bd4-2f8d-4a86-9233-344ec6152f87]