

# Brian — PS 3 — 2025-01-24 — Solution

*EIWL3 Sections 9 and 10, and First Half of Section 11*

---

## Exercises from *EIWL3 Section 9*

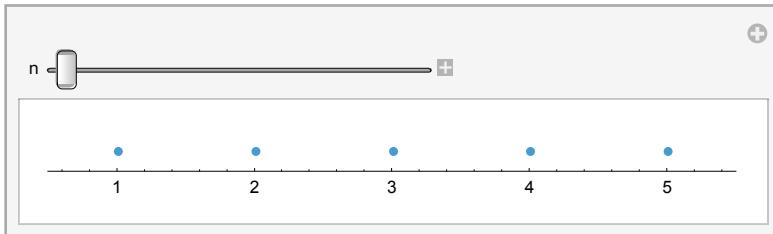
(\* 9.1 \*) Manipulate[Range[n], {n, 0, 100, 1}]

Out[*#*] =



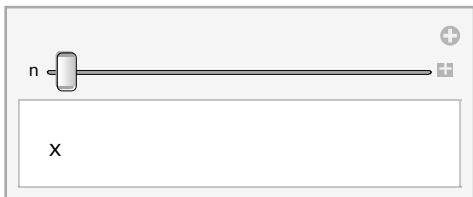
(\* 9.2 \*) Manipulate[NumberLinePlot[Range[n]], {n, 5, 50, 1}]

Out[*#*] =



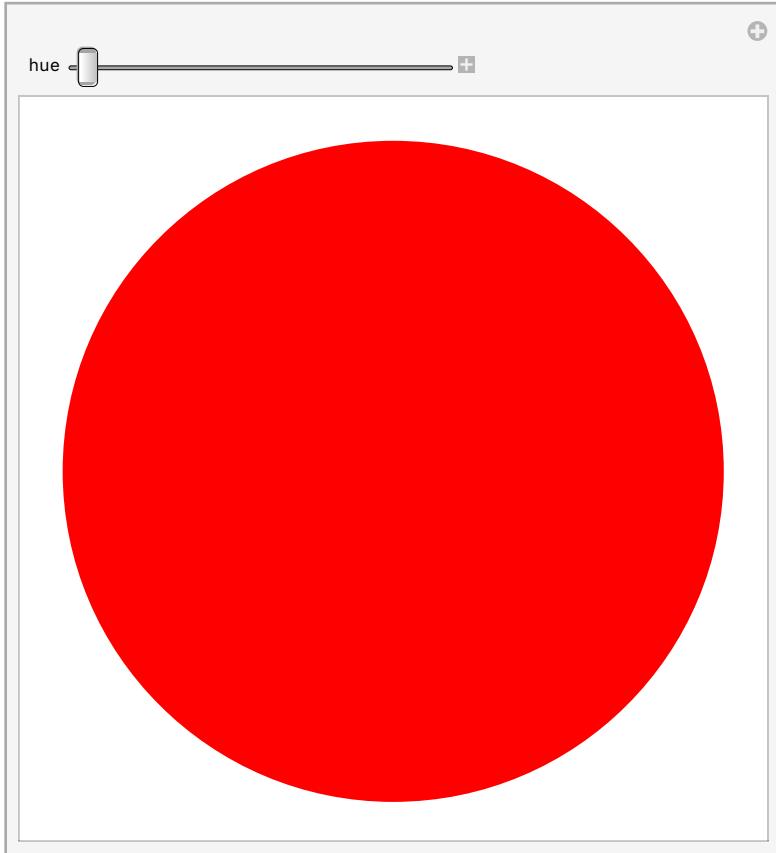
(\* 9.3 \*) Manipulate[Column[Table[x, n]], {n, 1, 10, 1}]

Out[*#*] =



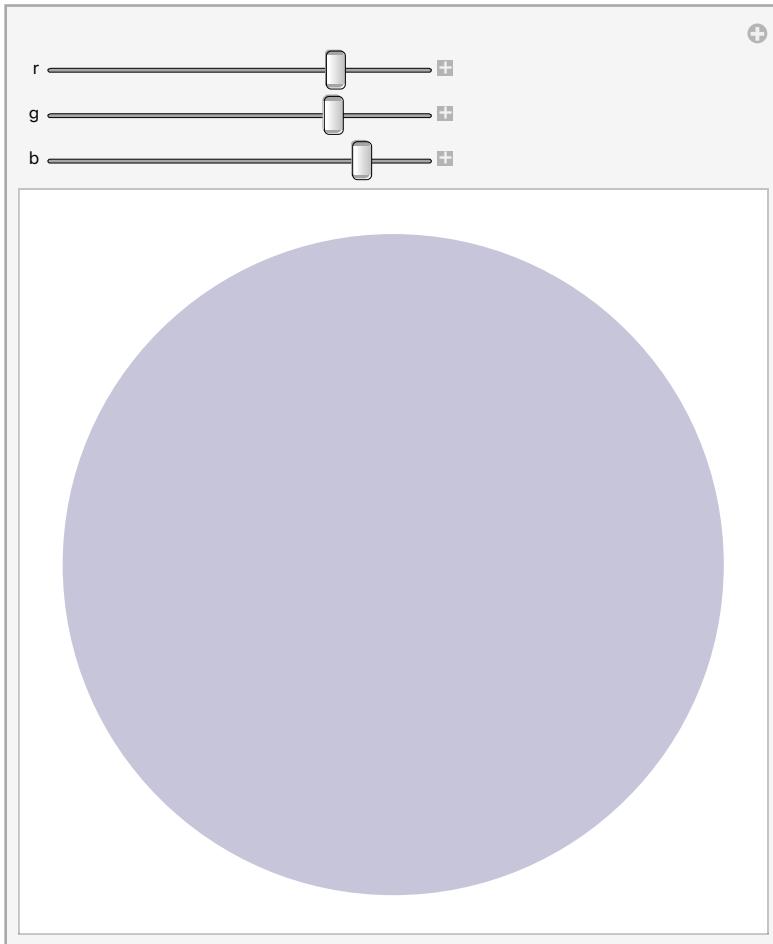
```
(* 9.4 *) Manipulate[Graphics[Style[Disk[], Hue[hue]], {hue, 0, 1}]
```

```
Out[=]=
```



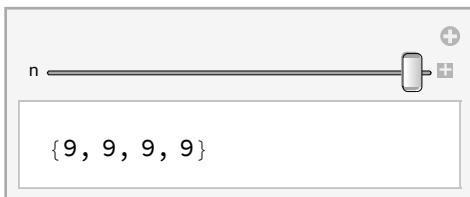
```
(* 9.5 *) Manipulate[  
  Graphics[Style[Disk[], RGBColor[r, g, b]]], {r, 0, 1}, {g, 0, 1}, {b, 0, 1}]
```

Out[•]=



```
(* 9.6 *) Manipulate[IntegerDigits[n], {n, 1000, 9999, 1}]
```

Out[•]=

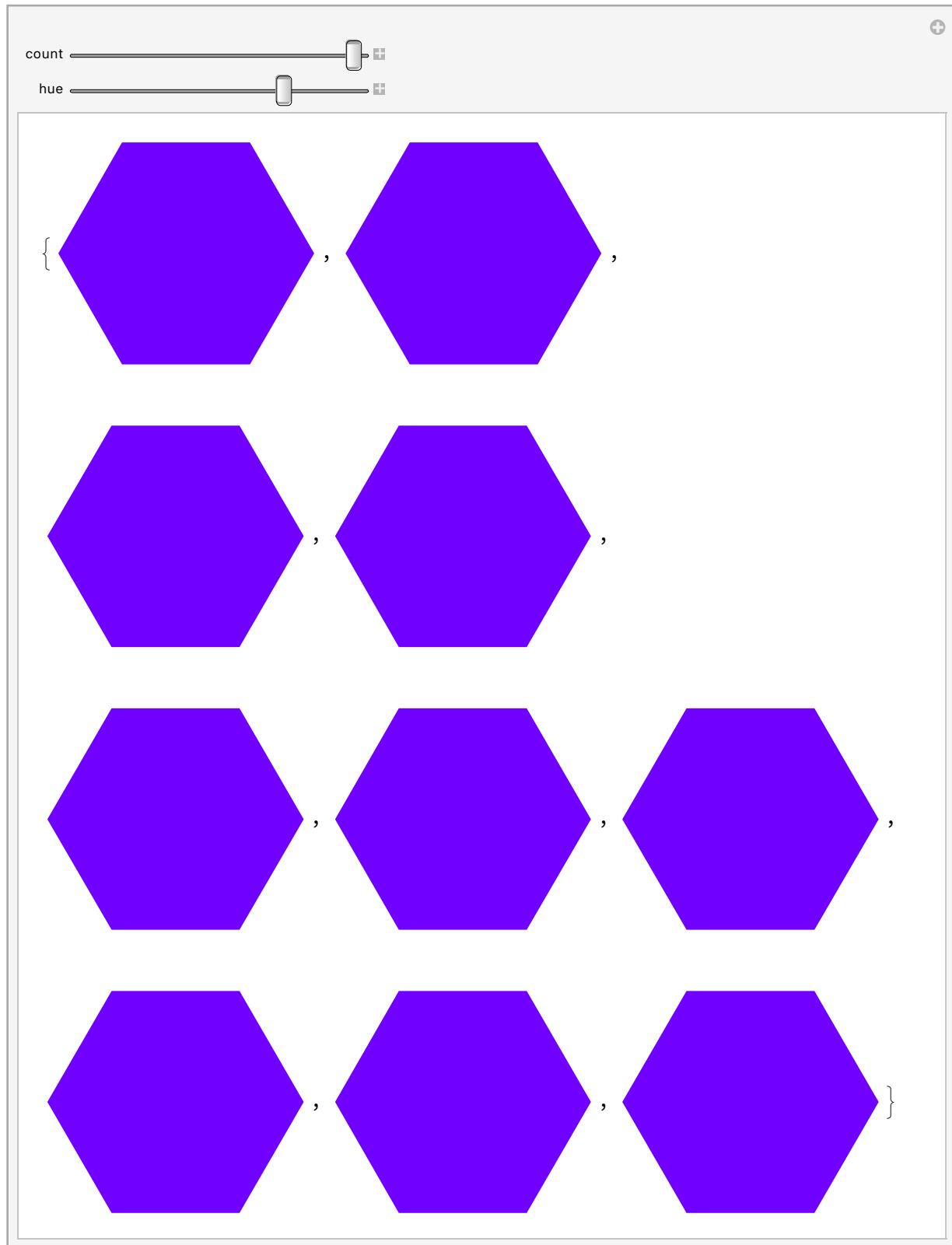


```
(* 9.7 *) Manipulate[Table[Hue[i/(n - 1)], {i, 0, n - 1}], {n, 5, 50}]  
(* Perhaps I got a bit carried away with my interpretation of equally-spaced :*)
```

Out[•]=

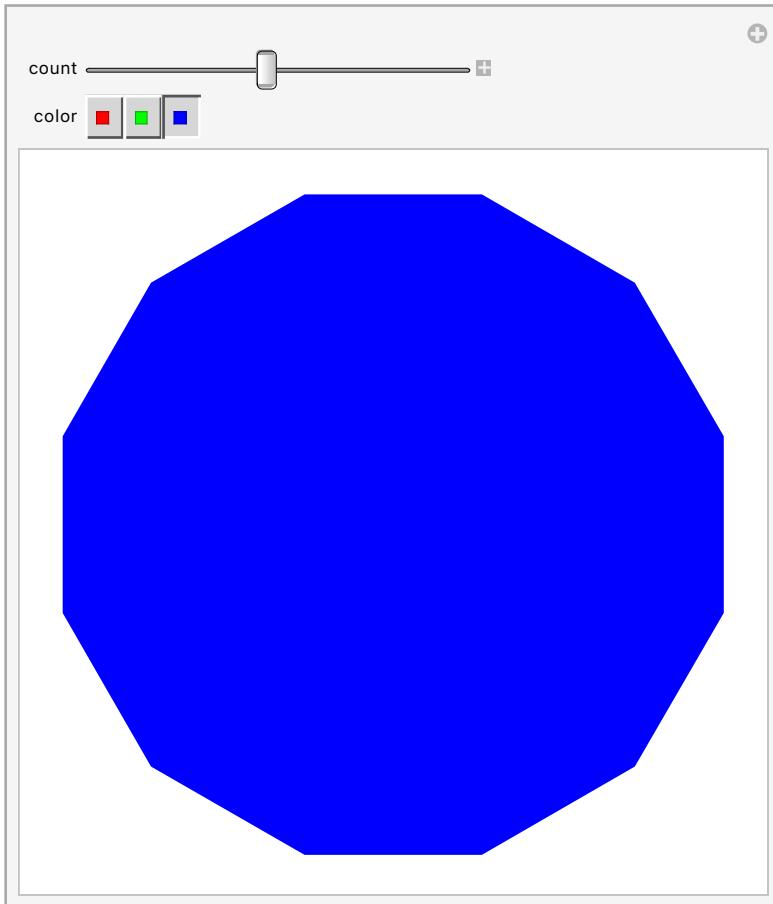


```
(* 9.8 *) Manipulate[Table[Graphics[Style[RegularPolygon[6], Hue[hue]]], count],  
{count, 1, 10, 1}, {hue, 0, 1}]
```

Out[*o*]=

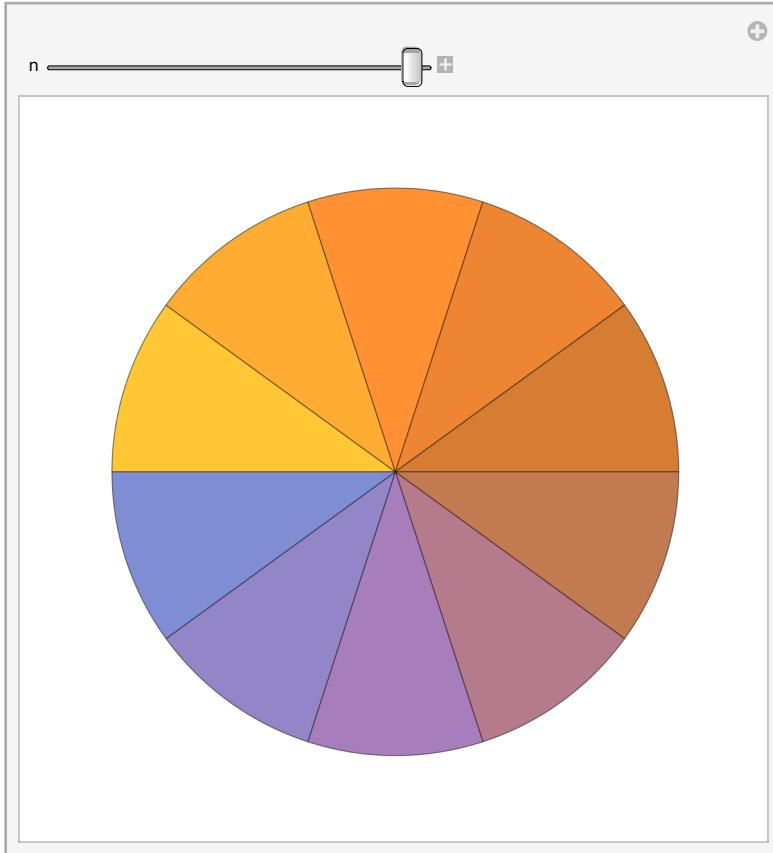
```
(* 9.9 *) Manipulate[Graphics[Style[RegularPolygon[count], color]],  
{count, 5, 20, 1}, {color, {Red, Green, Blue}}]
```

Out[•]=



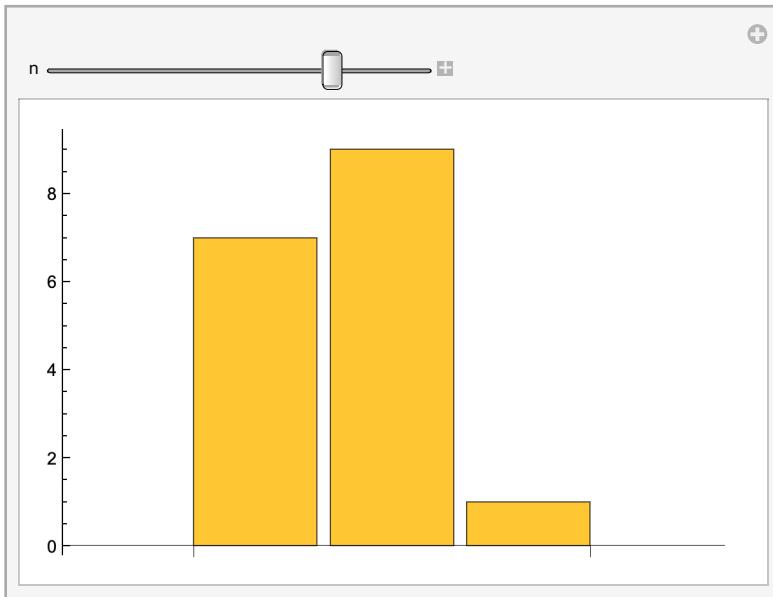
```
(* 9.10 *) Manipulate[PieChart[Table[1, n]], {n, 1, 10}]
```

Out[ ]=



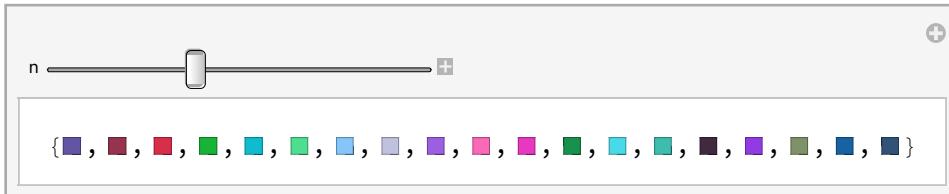
```
(* 9.11 *) Manipulate[BarChart[IntegerDigits[n]], {n, 100, 999, 1}]
```

Out[ ]=



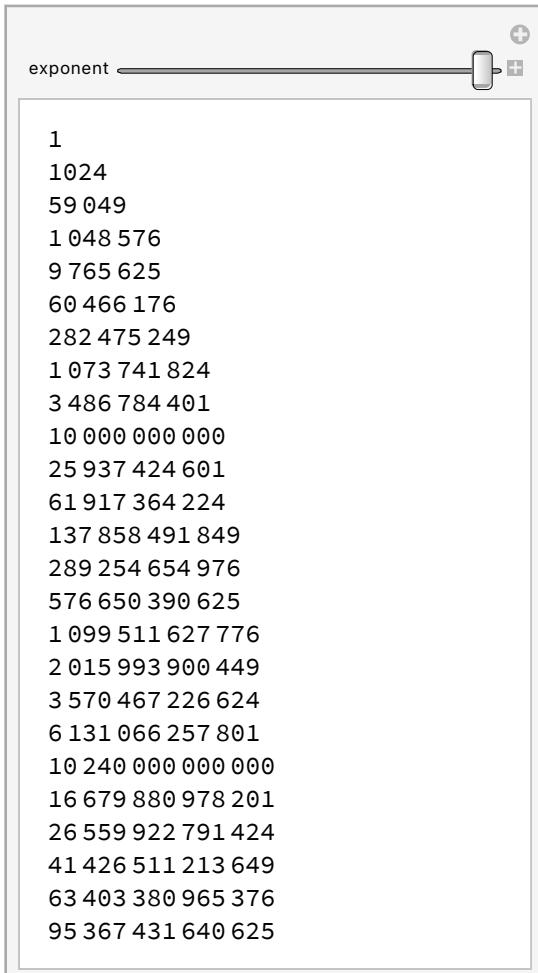
```
(* 9.12 *) Manipulate[Table[RandomColor[], n], {n, 1, 50}]
```

Out[•]=



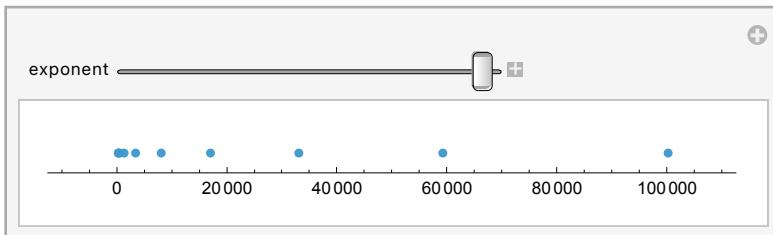
```
(* 9.13 *) Manipulate[Column[Table[base^exponent, {base, 1, 25}]], {exponent, 1, 10, 1}]
```

Out[•]=



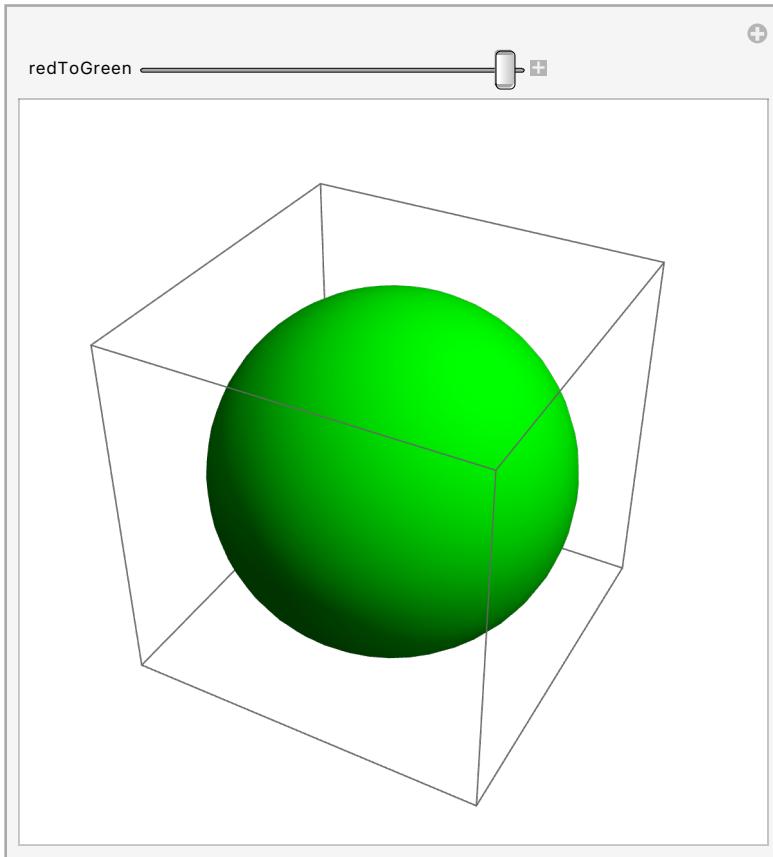
```
(* 9.14 *) Manipulate[NumberLinePlot[Range[10]^exponent], {exponent, 0, 5, 1}]
```

Out[•]=



```
(* 9.15 *) Manipulate[Graphics3D[  
  Style[Sphere[], RGBColor[1 - redToGreen, redToGreen, 0]]], {redToGreen, 0, 1}]
```

Out[•]=



---

## Exercises from EIWL3 Section 10

```
anImage = CurrentImage[];  
(* An image I will be re-using for lots of the exercises *)
```

```
(* 10.1 *) ColorNegate[anImage]
```

Out[*n*] =



```
In[n] := (* 10.2 *) Manipulate[Blur[anImage, blur], {blur, 0, 20}]
```

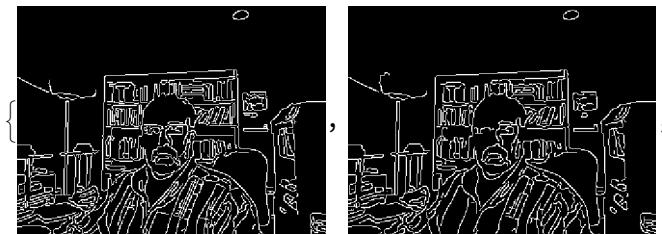
Out[*n*] =

A Manipulate interface for the Blur function. It shows a slider labeled "blur" ranging from 0 to 20, and the input field "Blur[anImage, 0.]" which is highlighted with a red border.

••• **Blur**: Expecting an image or graphics instead of anImage.

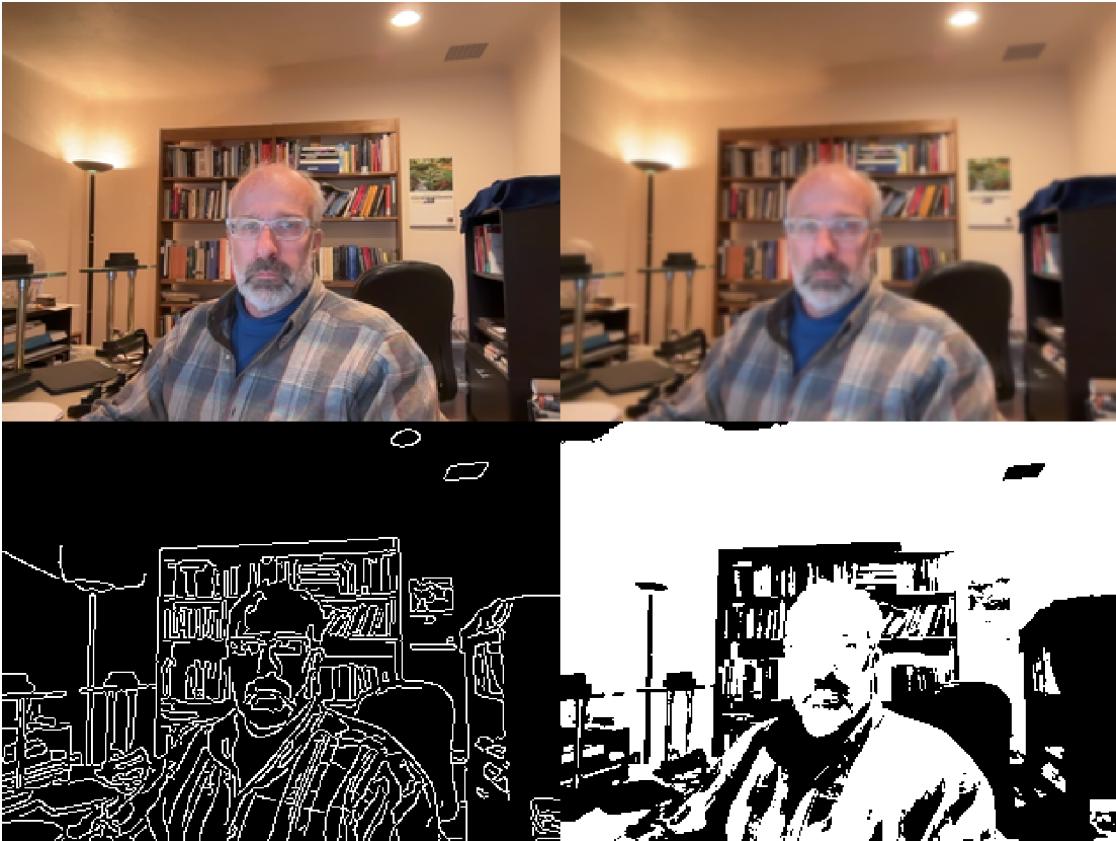
```
In[=]:= (* 10.3 *) Table[EdgeDetect[Blur[anImage, blur]], {blur, 1, 10}]
```

```
Out[=]=
```



```
In[=]:= (* 10.4 *)
ImageCollage[{anImage, Blur[anImage], EdgeDetect[anImage], Binarize[anImage]}]
```

Out[=]=



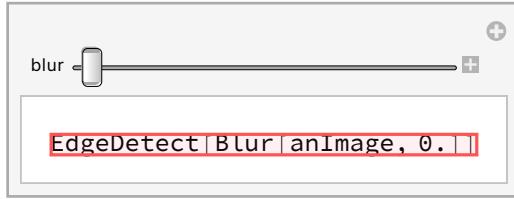
```
(* 10.5 *) anImage + Binarize[anImage]
```

Out[=]=



```
(* 10.6 *) Manipulate[EdgeDetect[Blur[anImage, blur]], {blur, 0, 20}]
```

Out[ ]=

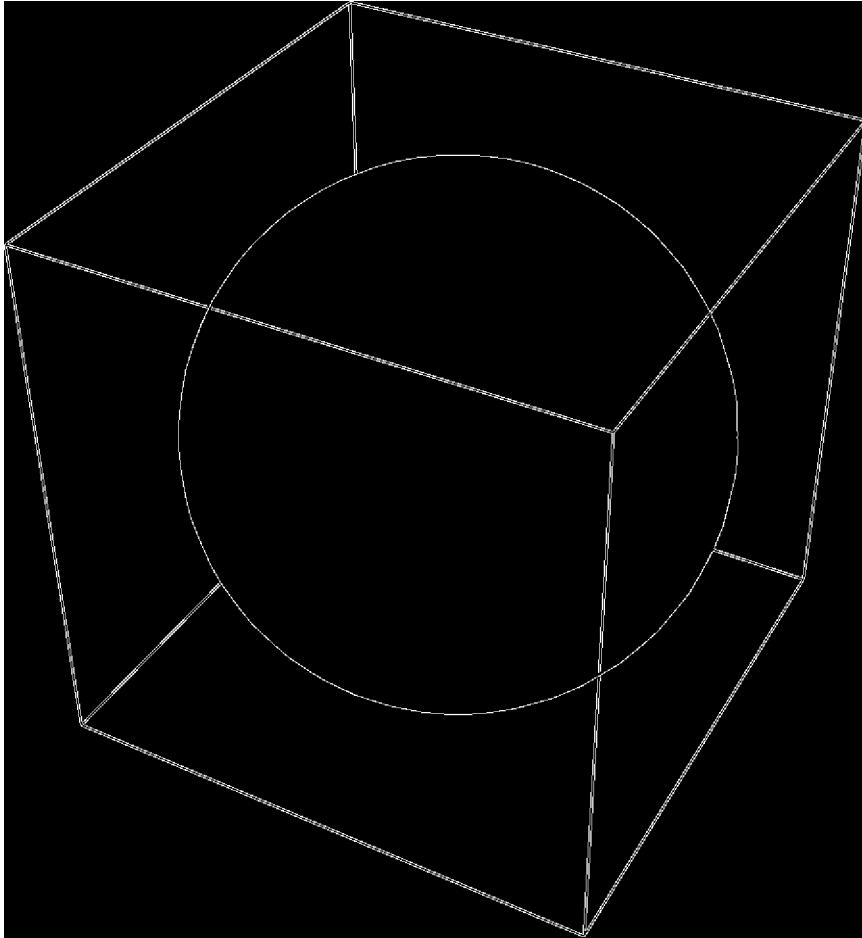


••• **Blur**: Expecting an image or graphics instead of anImage.

••• **EdgeDetect**: Expecting an image or graphics instead of Blur[anImage, 0.].

```
In[ ]:= (* 10.7 *) EdgeDetect[Graphics3D[Sphere[]]]
```

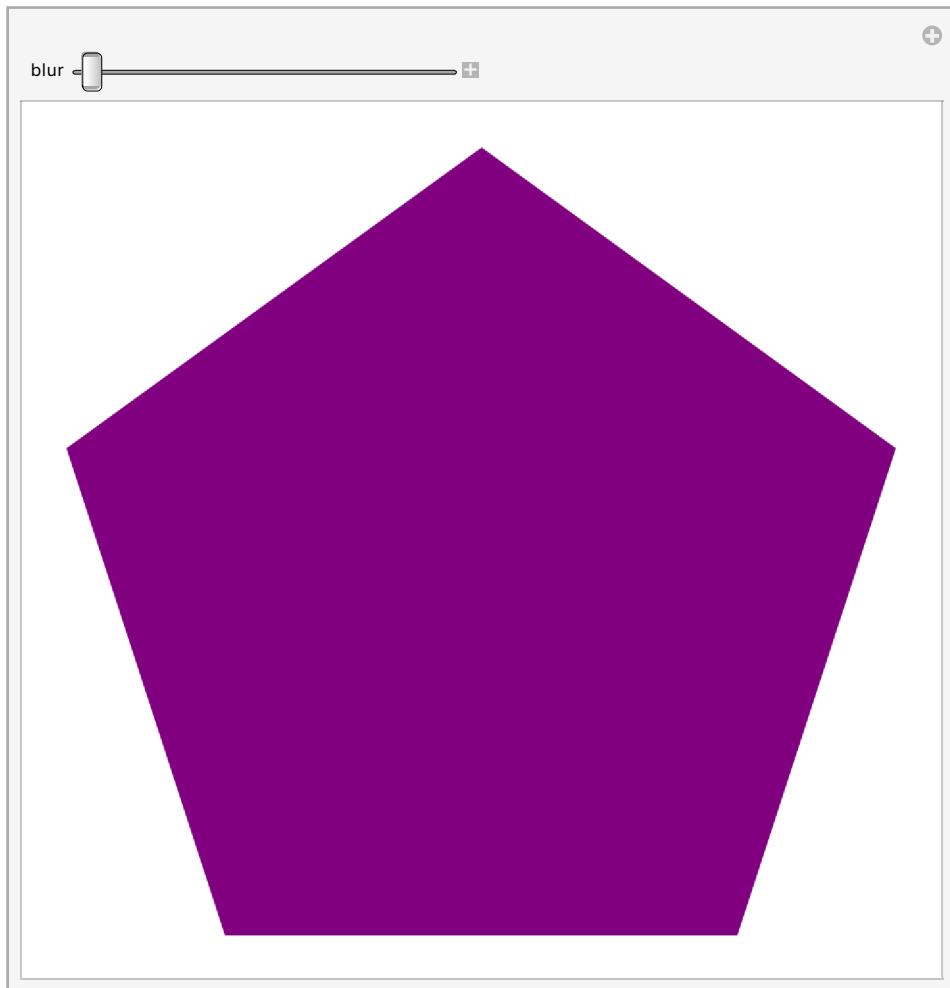
Out[ ]=



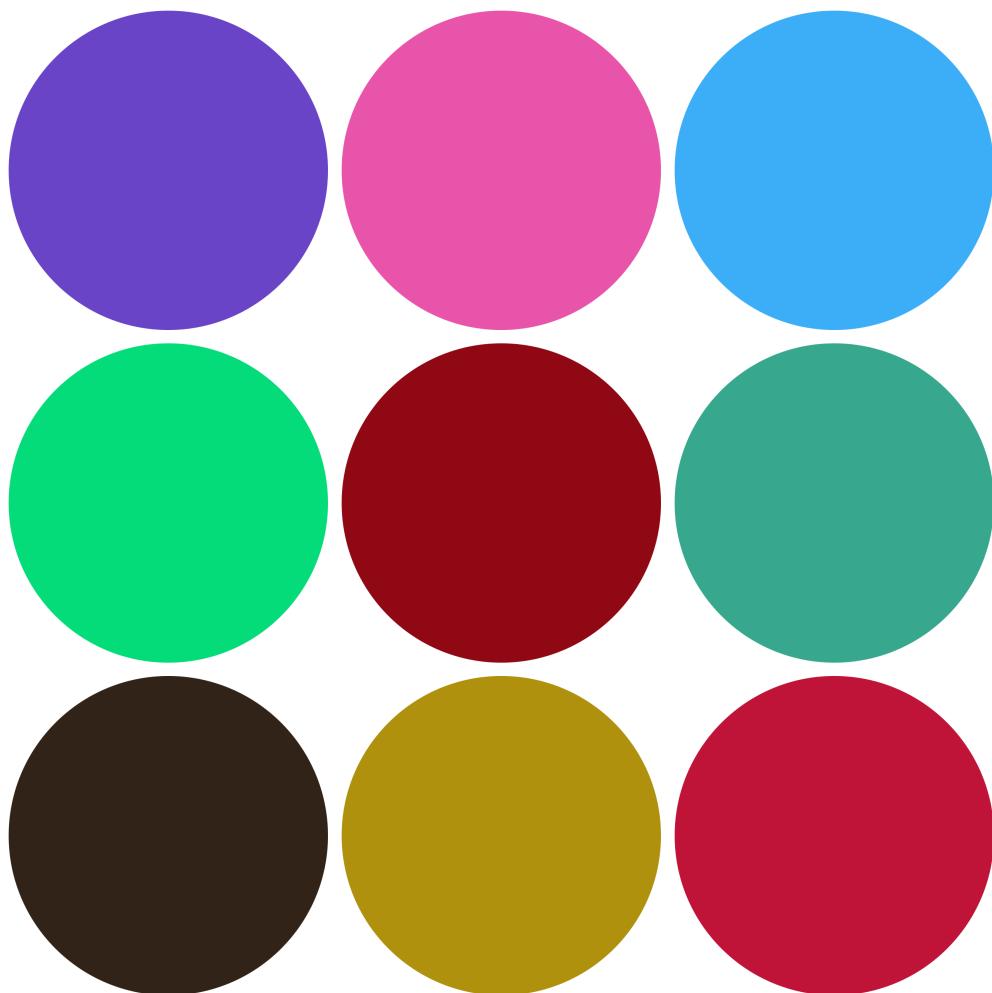
```
In[6]:= (* 10.8 *)
```

```
Manipulate[Blur[Graphics[Style[RegularPolygon[5], Purple]], blur], {blur, 0, 20}]
```

```
Out[6]=
```

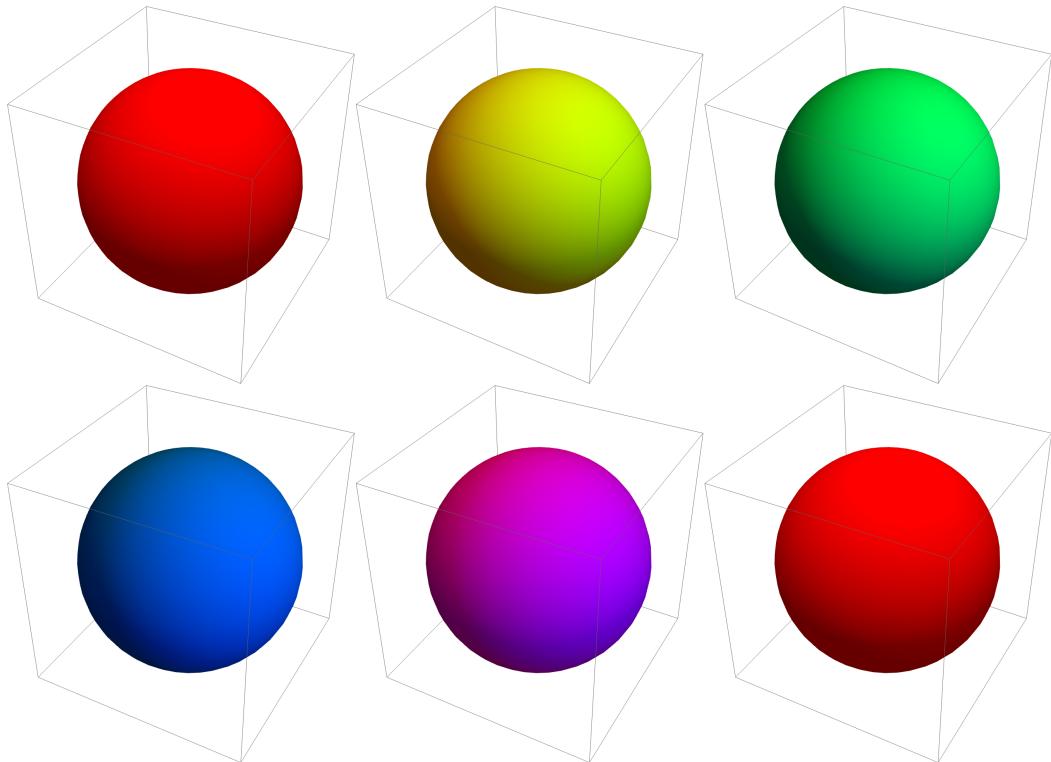


```
In[6]:= (* 10.9 *) ImageCollage[  
Table[  
Graphics[Style[Disk[], RandomColor[]]],  
{i, 1, 9}]]  
]  
Out[6]=
```



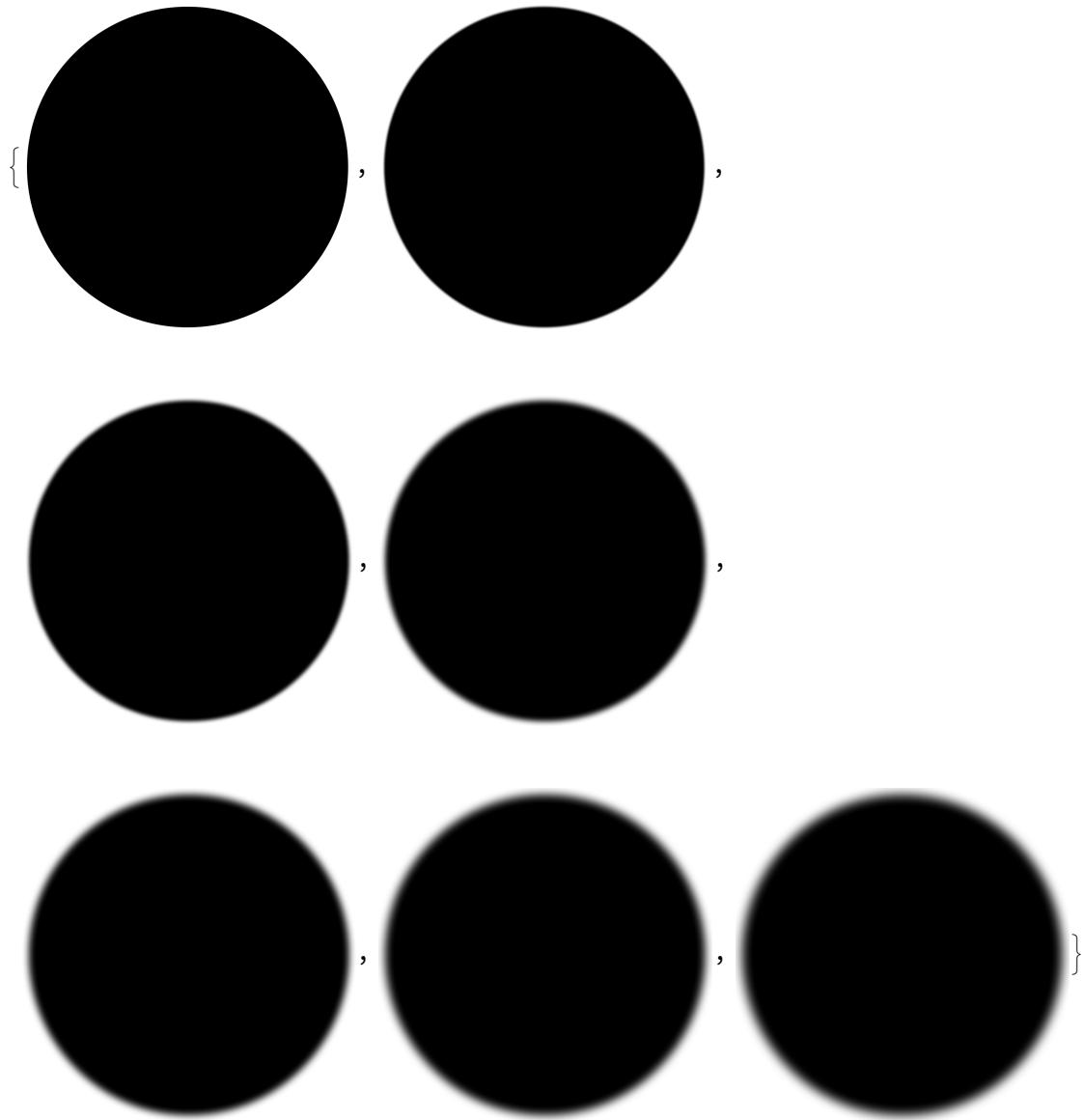
```
In[6]:= (* 10.10 *) ImageCollage[
Table[
Graphics3D[Style[Sphere[], Hue[hue]]],
{hue, 0.0, 1.0, 0.2}]
]
```

```
Out[6]=
```



```
In[6]:= (* 10.11 *) Table[
  Blur[Graphics[Disk[]], blur],
  {blur, 0, 30, 5}]
```

Out[6]=



```
In[=]:= (* 10.12 *) ImageAdd[anImage, Graphics[Disk[]]]  
Out[=]=
```



```
In[=]:= (* 10.13 *) ImageAdd[anImage, Graphics[Style[RegularPolygon[8], Red]]]  
Out[=]=
```



```
In[=]:= (* 10.14 *) ImageAdd[anImage, ColorNegate[EdgeDetect[anImage]]]  
Out[=]=
```



## Exercises 11.1-11.15 from EWL3 Section 11

```
In[1]:= (* 11.1 *) StringJoin[Table["Hello", 2]]
Out[1]= HelloHello

(* 11.2 *) ToUpperCase[StringJoin[Alphabet[]]]
Out[2]= ABCDEFGHIJKLMNOPQRSTUVWXYZ

(* 11.3 *) StringJoin[Reverse[Alphabet[]]]
Out[3]= zyxwvutsrqponmlkjihgfedcba

(* 11.4 *) StringJoin[Reverse[Alphabet[]]]
Out[4]= zyxwvutsrqponmlkjihgfedcba

In[5]:= (* 11.5 *) StringTake[StringJoin[Alphabet[]], 6]
Out[5]= abcdef

(* 11.6 *) (* I had to look up a solution for this,
mostly because I did not understand the question. *)
Column[Table[StringTake["this is about strings", n],
{n, StringLength["this is about strings"]}],
```

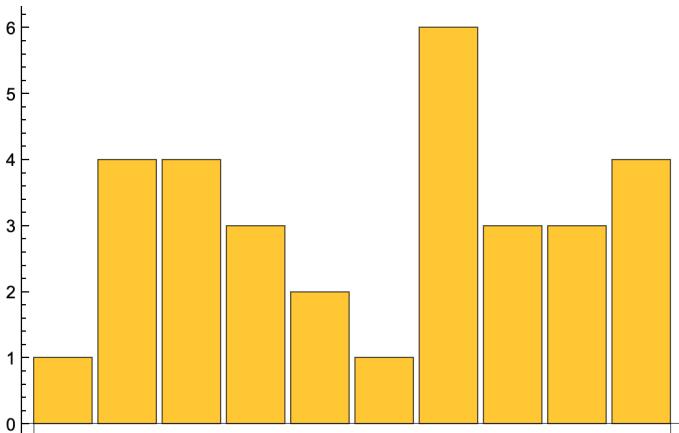
Out[6]=

```
t
th
thi
this
this
this i
this is
this is
this is a
this is ab
this is abo
this is abou
this is about
this is about
this is about s
this is about st
this is about str
this is about stri
this is about strin
this is about string
this is about strings
```

(\* 11.7 \*)

BarChart[StringLength[TextWords["A long time ago, in a galaxy far, far away"]]]

Out[•]=



(\* 11.8 \*) StringLength[WikipediaData["Computer"]]

Out[•]=

60266

In[•]:= (\* 11.9 \*) Length[TextWords[WikipediaData["Computer"]]]

Out[•]=

9271

In[•]:= (\* 11.10 \*) First[TextSentences[WikipediaData["Computer"]]]

Out[•]=

A computer is a machine that can be programmed to automatically carry out sequences of arithmetic or logical operations (computation).

In[•]:= (\* 11.11 \*) StringJoin[StringTake[TextSentences[WikipediaData["Computer"]], 1]]

Out[•]=

```
AMTTACCESEMTTCTPP=ITTDDBTTTT==DTLTTTSITIIDMTAAATTIASIBIAITITIITSI=CCAHTFTTAEBNH
=ITax()2{,THI=DHTTTTAB==CBTDETTITIRTZTT=PTEITDTTHACIINCTLOTIIIHBT==TTHTVTE=
ECWATIHJTIIAATBAIL=TJFCJTHATHTTWITT=TTDTKIHKNHPNIMTGFTTWISTITS=TTLTTTT=C=A=SH=
TC==ATIET=WTTSC=TSC=TCATRDIRTPIWJSAIT=TES=TTSHTALTSG=
AETTLSIETWOAMTTRACrRIISFIIG=IDOHCIAAMA=WTOBITSTBSIT=SMSTSS=SSCICW=T=TTMIAL=
TITHTFMPWSTCBOTOI=ITTSTTITMWITC=PUTTS=MF=ATHHIT=PALTP=ETHOBSA=CTITTCITA"=AWA=
TMH=TQCVSLTTT=ACARPE=AT=====M
```

(\* 11.12 \*) Last[SortBy[WordList[], StringLength]]

(\* I learned while grading that using Max is better than what I did \*)

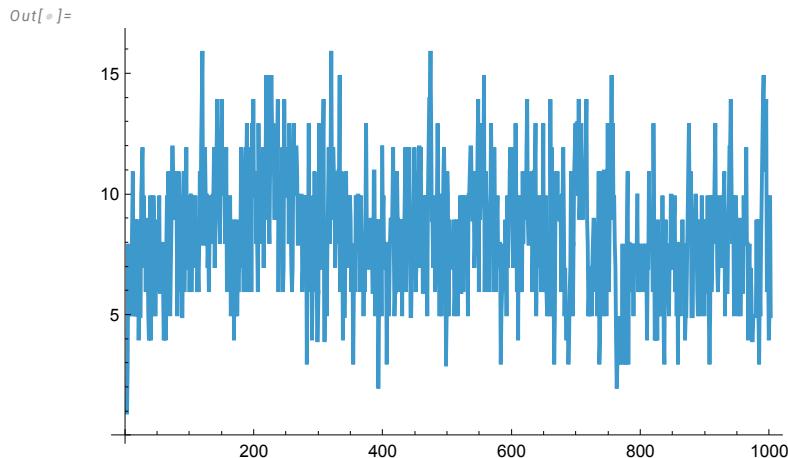
Out[•]=

electroencephalographic

```
(* 11.13 *) (* I had to look up a solution for this,  
but now that I see it, I should have had it on my own. *)  
Count[StringTake[WordList[], 1], "q"]
```

Out[\*]=  
194

```
In[*]:= (* 11.14 *) ListLinePlot[StringLength[Take[WordList[], 1000]]]
```



```
(* 11.15 *) WordCloud[Characters[StringJoin[WordList[]]]]
```

