Pensieve header: October 30: Textbook (EIWL) chapters 9-12 (plus Dynamic).

Today. EIWL-9-12, then maybe **Patterns**.

Topics (in no particular order). Whatever you may suggest; whatever comes to my mind; ~~the Fibonacci numbers~~; the Catalan numbers; ~~the Jones polynomial~~; ~~a more efficient Jones algorithm~~; ~~a riddle on spheres~~; Khovanov homology; Γ-calculus; the Hopf fibration; Hilbert's 13th problem; non-commutative Gaussian elimination; free Lie algebras; the Baker-Campbell-Hausdorff formula; wacky numbers; an order 4 torus; the Schwarz Lantern; knot colourings; the Temperley-Lieb pairing; the dodecahedral link; sound experiments; barycentric subdivisions; ~~some Peano curves~~; braid closures and Vogel's algorithm; the insolubility of the quintic; phase portraits; the Mandelbrot set; shadows of the Cantor aerogel; quilt plots; some image transformations; De Bruijn graphs; the Riemann series theorem; finite type invariants and the Willerton fish; ~~the Towers of Hanoi~~; ~~Hochschild homology of (some) coalgebras~~; convolutions and image improvements.

An Image Manipulation Challenge

The image at http://drorbn.net/bbs/show?shot=17-1750-171016-111042.jpg is pathetic. Can you improve it? Whatever you do, should also work well with all other images at http://drorbn.net/bbs/show.php?prefix=17-1750.

A Graphics Challenge

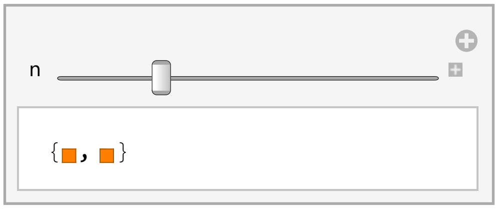
The torus $S^1 \times S^1$ has an order 4 symmetry. Can you draw it in such a manner that it will manifest?

New Rule

Submissions are limited to 20Mb.

9. Interactive Manipulation

```
Manipulate[Table[Orange, n], {n, 1, 5, 1}]
```

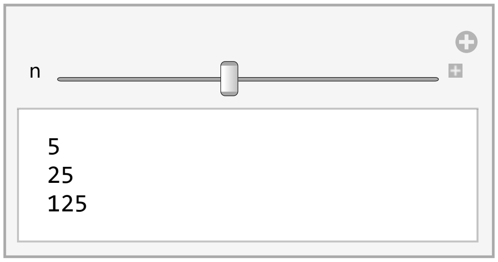


```
{🟧, 🟧}
```

```
Table[Table[Orange, n], {n, 1, 5, 1}]
```

```
{{🟧}, {🟧, 🟧}, {🟧, 🟧, 🟧}, {🟧, 🟧, 🟧, 🟧}, {🟧, 🟧, 🟧, 🟧, 🟧}}
```

```
Manipulate[Column[{n, n^2, n^3}], {n, 1, 10, 1}]
```

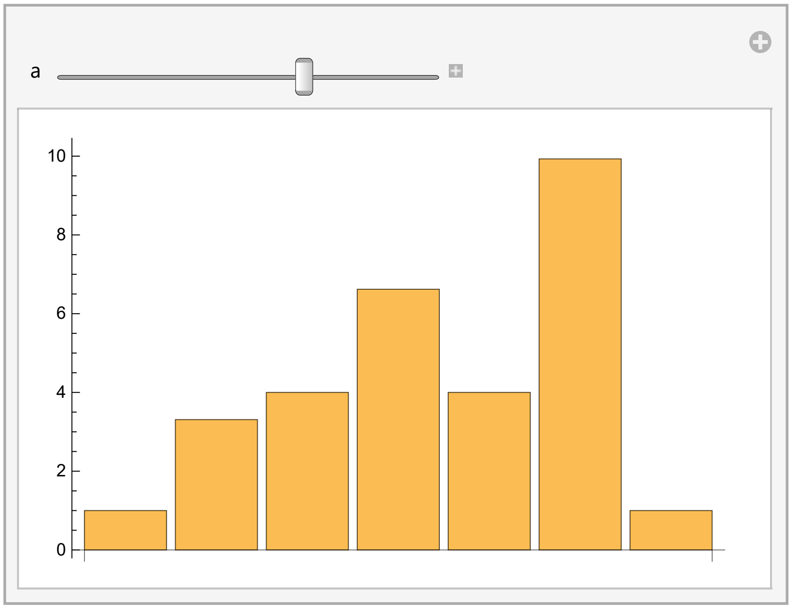


```
5
25
125
```

```
Table[Column[{n, n^2, n^3}], {n, 1, 10, 1}]
```

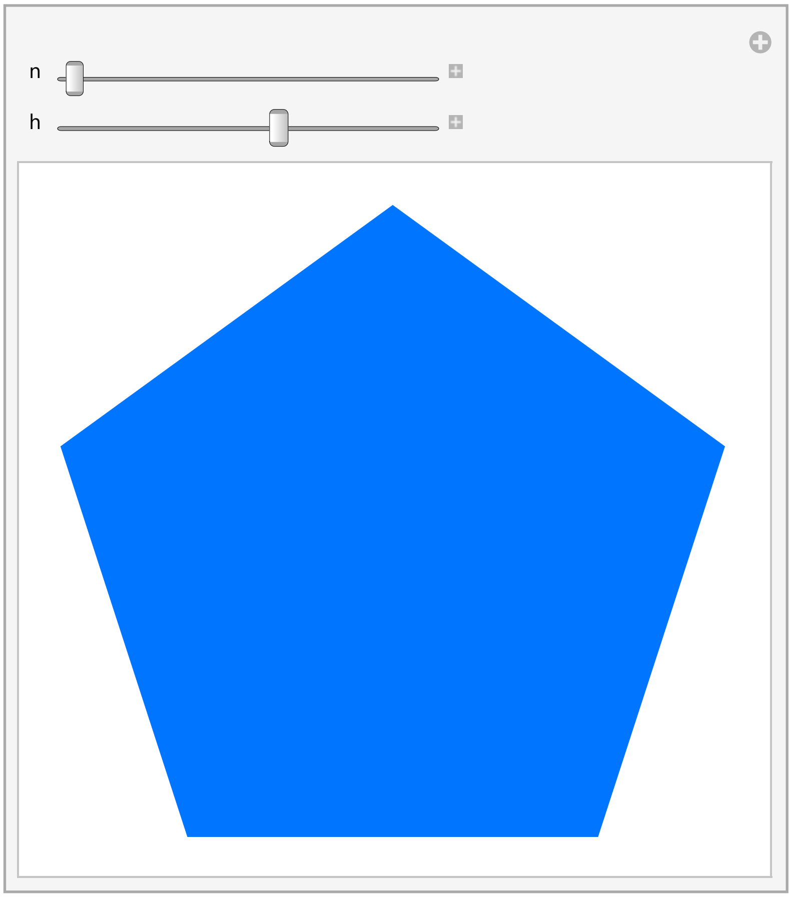
```
Manipulate[Column[{n, n^2, n^3}], {n, 1, 10}]
```

```
Manipulate[BarChart[{1, a, 4, 2 * a, 4, 3 * a, 1}], {a, 0, 5}]
```

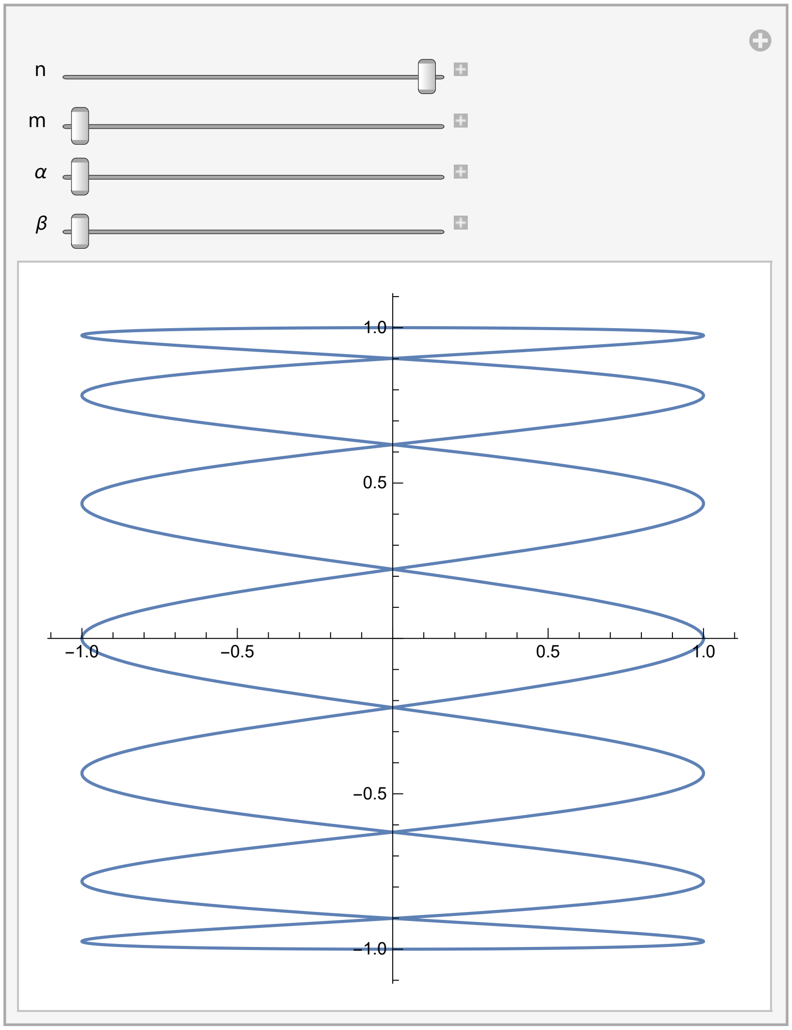


```
Manipulate[PieChart[{1, a, 4, 2 * a, 4, 3 * a, 1}], {a, 0, 5}]
```

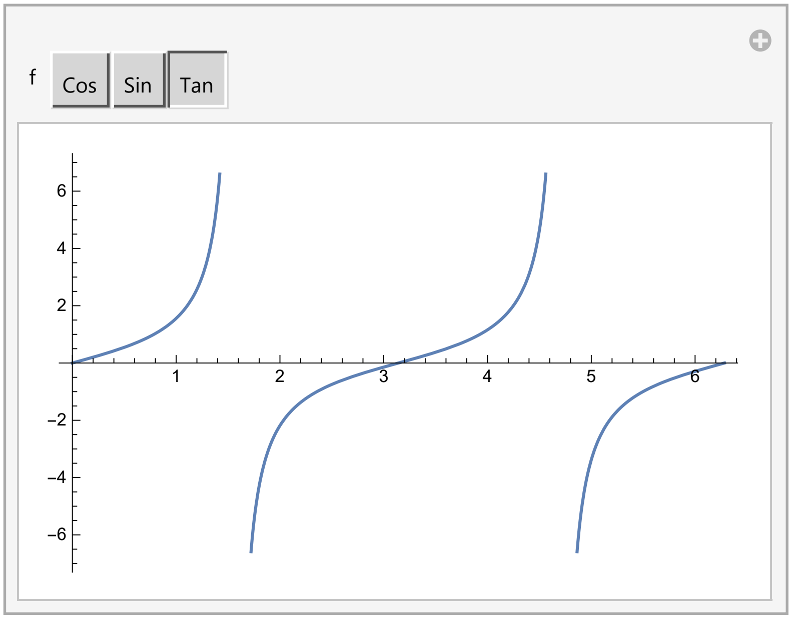
```
Manipulate[Graphics[Style[RegularPolygon[n], Hue[h]]], {n, 5, 20, 1}, {h, 0, 1}]
```



```
Manipulate[
 ParametricPlot[{Cos[α + n t], Sin[β + m t]}, {t, 0, 2 π}],
 {n, 1, 7, 1}, {m, 1, 7, 1}, {α, 0, 2 π / n}, {β, 0, 2 π}
]
```



```
Manipulate[Plot[f[x], {x, 0, 2 π}], {f, {Cos, Sin, Tan}}]
```



```
Manipulate[Graphics[Style[RegularPolygon[5], color]], {color, {Red, Yellow, Blue}}]
```

```
? Dynamic
```

```
Dynamic[expr] represents an object that displays as the dynamically updated current value of
   expr. If the displayed form of Dynamic[expr] is interactively changed or edited, an assignment
   expr = val is done to give expr the new value val that corresponds to the displayed form.
Dynamic[expr, None] does not allow interactive changing or editing.
Dynamic[expr, f] continually evaluates f[val, expr] during interactive changing or editing of val.
Dynamic[expr, {f, f_end}] also evaluates f_end[val, expr] when interactive changing or editing is complete.
Dynamic[expr, {f_start, f, f_end}] also evaluates f_start[val, expr] when interactive changing or editing begins.  ≫
```

```
Dynamic[x]
```

```
0.
```

```
x = 7
```

```
7
```

```
? Prime
```

```
Prime[n] gives the n^th prime number.  ≫
```

```
PrimePi[10^10]
```

```
455052511
```

```
PrimePi[100]
```

```
25
```

```
? PrimeQ
```

```
PrimeQ[expr] yields True if expr is a prime number, and yields False otherwise.  ≫
```

```
Dynamic[n]
```

```
n
```

```
primes = 0;
Do[
 prime = True; k = 2;
 While[prime ∧ k^2 ≤ n, If[IntegerQ[n / k], prime = False, ++k]];
 If[prime, ++primes],
 {n, 2, 10^10}
];
primes
```

```
$Aborted
```

```
? Slider
```

```
Slider[x] represents a slider with setting x in the range 0 to 1.
Slider[Dynamic[x]] takes the setting to be the dynamically
   updated current value of x, with the value of x being reset if the slider is moved.
Slider[x, {x_min, x_max}] represents a slider with range x_min to x_max.
Slider[x, {x_min, x_max, dx}] represents a slider that jumps in steps dx.
Slider[x, {{e_1, e_2, …}}] represents a slider
   in which equally spaced intervals correspond to successive settings e_i.
Slider[x, {{{e_1, w_1}, {e_2, w_2}, …}}] uses intervals of relative widths w_i for the e_i.  ≫
```

```
Clear[x]
```

```
Slider[Dynamic[x]]
```

```
Dynamic[x]
```

```
0.
```

```
? LocatorPane
```

LocatorPane[{*x*, *y*}, *back*] represents a pane with a locator at position {*x*, *y*} and background *back*.

LocatorPane[Dynamic[*pt*], *back*] takes the locator position to be the dynamically updated current value of *pt*, with the value of *pt* being reset if the locator is moved.

LocatorPane[{pt_1, pt_2, …}, *back*] sets up multiple locators at positions pt_1, pt_2, ….

LocatorPane[Dynamic[{pt_1, pt_2, …}], *back*] takes the locator positions to be dynamically updated current values of the pt_i.

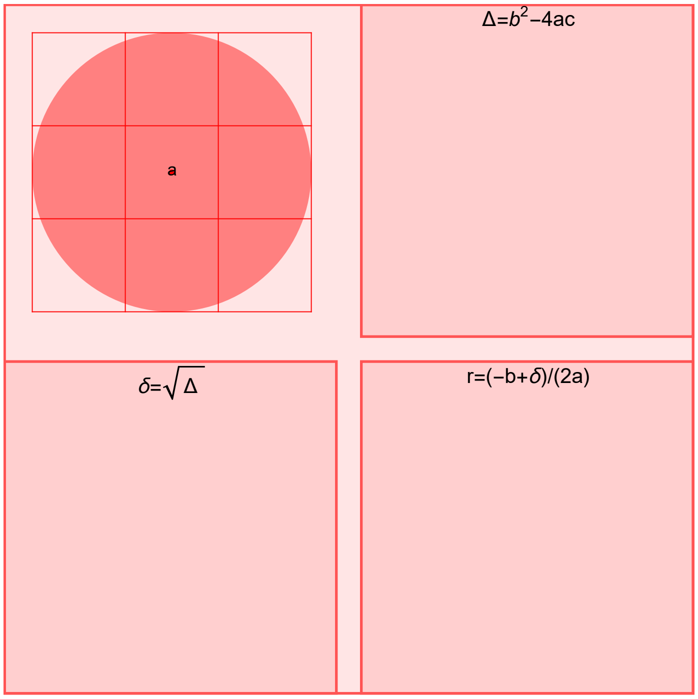
LocatorPane[*pts*, *back*, {{x_{min}, y_{min}}, {x_{max}, y_{max}}}] specifies the range of coordinates for the locator.

LocatorPane[*pts*, *back*, {{x_{min}, y_{min}}, {x_{max}, y_{max}}, {*dx*, *dy*}}] uses jumps *dx*, *dy*. ≫

From “Nobody Solves the Quintic”, http://www.math.toronto.edu/~drorbn/Talks/Sydney-1708/:

```
InputBackground = Graphics[{
    Pink, Disk[],
    Red, Point[{0, 0}],
    Table[{Line[{{t, -1}, {t, 1}}], Line[{{-1, t}, {1, t}}]}, {t, -1, 1, 2/3}]
   }];
OutputBackground = {
   LightBlue, Disk[],
   Blue, Point[{0, 0}],
   Table[{Line[{{t, -1}, {t, 1}}], Line[{{-1, t}, {1, t}}]}, {t, -1, 1, 2/3}],
   Black
  };
```

```
Module[{a0, b0, c0, a, b, c, △, δ, r},
 {a0, b0, c0} = {{1, 0}, {0, 0}, {0, 1/3}};
GraphicsGrid[Partition[#, 2] & @{
    LocatorPane[Dynamic[{a0, b0, c0}], InputBackground, Appearance → {"a", "b", "c"}],
    Dynamic[Graphics[{OutputBackground,
       a = {1, i}.a0; b = {1, i}.b0; c = {1, i}.c0;
       △ = b^2 - 4 a c; Text["△", ReIm@△]
      }, PlotRange → All, PlotLabel → "△=b^2-4ac"]],
    Dynamic[Graphics[{OutputBackground,
       δ = √△; Text["δ", ReIm@δ]
      }, PlotRange → All, PlotLabel → "δ=√△"]],
    Dynamic[Graphics[{OutputBackground,
       r = (-b + δ) / (2 a); Point[ReIm@r]
      }, PlotRange → All, PlotLabel → "r=(-b+δ)/(2a)"]]
   }]
]
```



```
$nsqrt = 7;
nsqrt[x_] := $nsqrt = MinimalBy[√x {1, -1}, Abs[# - $nsqrt] &][[1]]
```

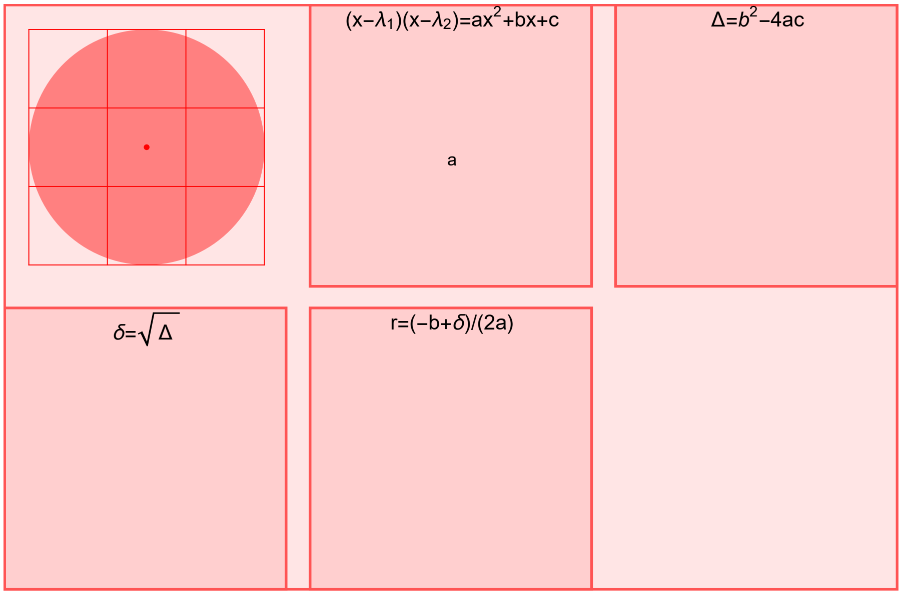
```
nsqrt[9]
```

```
3
```

```
nsqrt[-9]
```

```
3 i
```

```
Module[{λ1, λ2, x, a, b, c, Δ, δ, r},
 {λ1, λ2} = {{1, 1} / 2, {1, -1} / 2};
GraphicsGrid[Partition[#, 3] & @ {
    LocatorPane[Dynamic[{λ1, λ2}], InputBackground, Appearance → {"λ1", "λ2"}],
    Dynamic[Graphics[{OutputBackground,
       {c, b, a} = CoefficientList[(x - {1, ⅈ}.λ1) (x - {1, ⅈ}.λ2), x];
       Text["a", ReIm@a], Text["b", ReIm@b], Text["c", ReIm@c]
      }, PlotRange → All, PlotLabel → "(x-λ1) (x-λ2) =ax^2+bx+c"]],
    Dynamic[Graphics[{OutputBackground,
       Δ = b^2 - 4 a c; Text["Δ", ReIm@Δ]
      }, PlotRange → All, PlotLabel → "Δ=b^2-4ac"]],
    Dynamic[Graphics[{OutputBackground,
       δ = nsqrt[Δ]; Text["δ", ReIm@δ]
      }, PlotRange → All, PlotLabel → "δ=√Δ "]],
    Dynamic[Graphics[{OutputBackground,
       r = (-b + δ) / (2 a); Point[ReIm@r]
      }, PlotRange → All, PlotLabel → "r= (-b+δ) / (2a) "]],
    Null
   }]
]
```



10. Images

```
CurrentImage[]
```

```
$ImagingDevices
```

```
$ImagingDevice = $ImagingDevices[[2]];
img = CurrentImage[]
```

```
ColorNegate[img]
```

```
{Blur[img], Blur[img, 10]}
```

```
Table[Blur[img, n], {n, 0, 15, 5}]
```

```
ImageCollage[Table[Blur[img, n], {n, 0, 15, 5}]]
```

```
DominantColors[img]
```

```
Binarize[img]
```

```
Manipulate[Binarize[img, t], {t, 0, 1}]

DominantColors[Binarize[img]]

img1 = EdgeDetect[img]

ImageAdd[img, img1]

imgs = WikipediaData["knot theory", "ImageList"]

ImageCollage[Scaled[1] → imgs, Method → "ClosestPacking", Background → White]

cf = Import["http://drorbn.net/ap/Classes/17-1750-ShamelessMathematica/20170929_110340.jpg"]

EdgeDetect[cf]

faces = FindFaces[cf]

ImageTrim[cf, #] & /@ faces
```

11. Strings and Text

```
"This is a string."

StringLength["hello"]

StringReverse["hello"]

ToUpperCase["I'm coding in the Wolfram Language!"]

StringTake["this is about strings", 10]

StringLength[StringTake["this is about strings", 10]]

StringJoin["Hello", " ", "there!", " How are you?"]

{"apple", "banana", "strawberry"}

StringTake[{"apple", "banana", "strawberry"}, 2]

StringJoin[{"apple", "banana", "strawberry"}]

Characters["a string is made of characters"]

Sort[Characters["a string of characters"]]

InputForm[Sort[Characters["a string of characters"]]]

TextWords["This is a sentence. Sentences are made of words."]

StringLength[TextWords["This is a sentence. Sentences are made of words."]]

StringTake[WikipediaData["knot theory"], 100]

WordCloud[DeleteStopwords[WikipediaData["knot theory"]]]

Take[WordList[], 20]

WordCloud[StringTake[WordList[], 1]]

RomanNumeral[1988]

Table[RomanNumeral[n], {n, 20}]

ListLinePlot[Table[StringLength[RomanNumeral[n]], {n, 100}]]

IntegerName[56]

ListLinePlot[Table[StringLength[IntegerName[n]], {n, 100}]]
```

```
Alphabet[]
```

```
LetterNumber[{"a", "b", "x", "y", "z"}]
```

```
FromLetterNumber[{10, 11, 12, 13, 14, 15}]
```

```
Alphabet["Russian"]
```

```
Rasterize[Style["ABC", 100]]
```

```
EdgeDetect[Rasterize[Style["ABC", 100]]]
```

```
FromCharacterCode /@ Range[1000]
```

12. Sound

```
Sound[SoundNote["C"]]
```

```
Sound[{SoundNote["C"], SoundNote["C"], SoundNote["G"]}]
```

```
Sound[Table[SoundNote[RandomInteger[12], 0.1, "Violin"], 20]]
```

```
Play[Sin[440 × 2 Pi t], {t, 0, 1}]
```

```
Manipulate[{ℯ^f, Play[Sin[ℯ^f 2 Pi t], {t, 0, 1}, SampleRate → 100 000]},
 {{f, Log@440}, Log@10, Log@30 000} ]
```