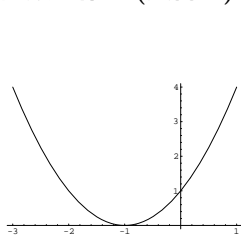
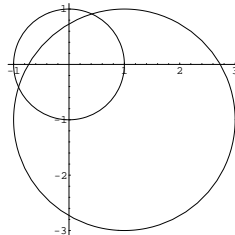


今回は、ソフトウェアのワークショップについて紹介します。Mathematicaを入試問題等の数値計算の確認には使ってはいるけれど、授業で活用するのは難しそうだと思っている先生も多いと思います。実は私もそんな一人だったのですが、「Mathematica 活用事例セミナー」(日本電子計算株式会社主催)に参加し、「Mathematicaを活用した教材例」に触れて、少しのコマンドやコツを覚えれば授業でも活用できると思いました。以下は、そのときの教材例です。

関数を作る (実習型)



例 1: 放物線のグラフ



例 2: 円のグラフ

生徒が各自で実習するための教材。生徒に入力させるコマンドを1つか2つに限って、パラメータ(数値)を変更させることで、スムーズな実習を行うことができます。

例 1: 放物線を定義する

```
fn2[a_, b_, c_] := Plot[ax^2+bx+c, {x, -b/2a-2, -b/2a+2}]
```

例 1: パラメータを代入する

```
fn2[1, 2, 1]
```

一度関数を作ると、簡単に使えるだけでなく、例2のように重ね合わせもできます。

例 2: 円を定義する

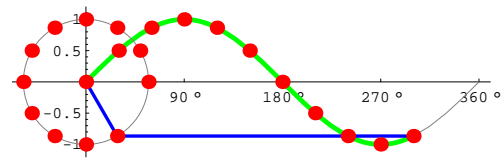
```
en[x_, y_, r_] := ParametricPlot[{r*Cos[θ]+x, r*Sin[θ]+y}, {θ, 0, 2π}, AspectRatio→Automatic, PlotRange→All]
```

例 2: パラメータを代入する

```
Show[{en[0, 0, 1], en[1, -1, 2]}]
```

グラフィックスの重ね合わせ (提示型)

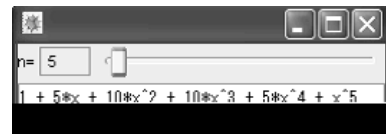
教員が生徒に対して提示するための教材。動く黒板的な教材が作れるので、黒板に瞬時に書くのが容易ではないグラフなどに向いています。Mathematicaのアニメーション機能とグラフィックスの重ね合わせを使います。

例 3: $y = \sin x$ のグラフ

```
Remove["Global`*"];
en=Graphics[{GrayLevel[0.5],Circle[{0,0},1]}];
g1=Plot[Sin[x],{x,0,2π},PlotStyle→{GrayLevel[0.5]},DisplayFunction→Identity];
g2[q_]:=Plot[Sin[x],{x,-0.001,q},PlotStyle→{Thickness[0.01],RGBColor[0,1,0]},DisplayFunction→Identity];
t1={};
ten1:=Graphics[{PointSize[0.03],RGBColor[1,0,0],Map[Point,t1]}];
t2={};
ten2:=Graphics[{PointSize[0.03],RGBColor[1,0,0],Map[Point,t2]}];
sen[q_]:=Graphics[{Thickness[0.007],RGBColor[0,0,1],Line[{0,0},{Cos[q],Sin[q]},q,Sin[q]]}];
Table[
t1=Append[t1,{Cos[q],Sin[q]}];
t2=Append[t2,{q,Sin[q]}];
Show[g1,g2[q],en,sen[q],ten1,ten2,AspectRatio→Automatic,
PlotRange→{-1.2,1.2},Ticks→{{90°,180°,270°,360°},Automatic},
DisplayFunction→$DisplayFunction],{q,0°,360°,30°}];
```

GUIkit を使った教材 (提示型, 実習型)

生徒にコマンドを入力させるのではなく、スライダーを動かさせることにより、変化する値や数式を表示させる教材。GUIkit というパッケージを使い、インターフェイスを作成するのに少し手間がかかりますが、直感的にわかりやすく操作性も良い教材が作れます。

例 4: $(x+1)^n$ の展開式

```
Needs["GUIkit"];
ref=GUIRun[Widget["Panel",{
Widget["Label",{text->"n="},
Widget["TextField",{text->"5",columns->3,editable->False},
Name->"textField",Widget["Slider",{value->5,BindEvent["change",
Script[{SetPropertyValue[{textField,"text"},
ToString[PropertyValue[{slider,"value"}],InputForm],
sliderChanged[]}]},
Name->"slider"]},
Widget["ScrollPane",{
"viewportView"->Widget["TextArea",{
"text"->"",rows->10,
"columns"->40,editable->False},Name->"textArea"]}],
Script[
sliderChanged[]:=Block[{n,x,y},
n=ToExpression[PropertyValue[{slider,"value"}]];
SetPropertyValue[{textField,"text"},
ToString[Expand[(x+1)^n],InputForm,PageWidth->
ToExpression[PropertyValue[{textArea,"columns"}]]
]];
sliderChanged[];
}]]];
```