

---

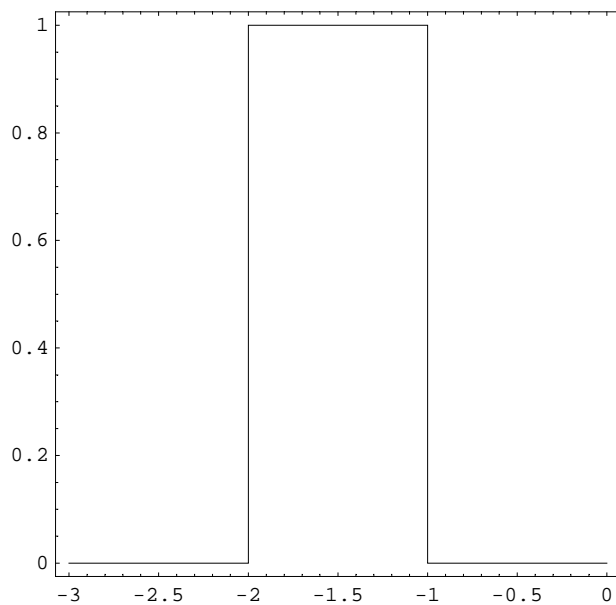
# Eine kleine Studie zu fraktalen Kurven

von Marco Wittwer und Josua Hönger

Freitag, 3. Dezember 2004

---

Auf der Homepage von Paul Bourke (<http://astronomy.swin.edu.au/~pbourke/>) sind diverse fraktale Grafiken mit ihrer zugehörigen Funktion abgebildet. Zum Beispiel die "Quadratic Snowflake":



Bei diesen Funktionen sind jeweils das Axiom (das Ausgangsmuster), F (das Ersetzungsmuster) und der Winkel angegeben. Für dieses Beispiel:

```
axiom = F  
F -> F-F+F+F-F  
angle = 90
```

Wir wollten diese Grafiken möglichst einfach nachbilden. Als erstes haben wir ein paar kleine Funktionen geschrieben.

```

stringreplace[s1_, s2_] := Module[{ns, i}, ns = {};
  For[i = 0, i < Length[s1], ns = Join[ns, Take[s1, {i, i}], s2], i++]; Return[ns];

makestring[Ax_, In_, It_] := Module[{i, S}, S = Ax;
  For[i = 0, i < It, S = stringreplace[S, In];
  i++];
  Return[S];];

plotstring[S_, Ratio_] := Module[{i, p, f, T}, p = {0, 0};
  T = Table[{0, 0}, {i, 1, Length[S] + 1}];
  T[[1]] = p;
  f = Pi / 2;
  For[i = 1, i ≤ Length[S], f += S[[i]];
  p = p + {Cos[f], Sin[f]};
  T[[i + 1]] = p;
  i++];
  ListPlot[T, PlotJoined → True, Frame → True, Axes → False, AspectRatio → Ratio]];

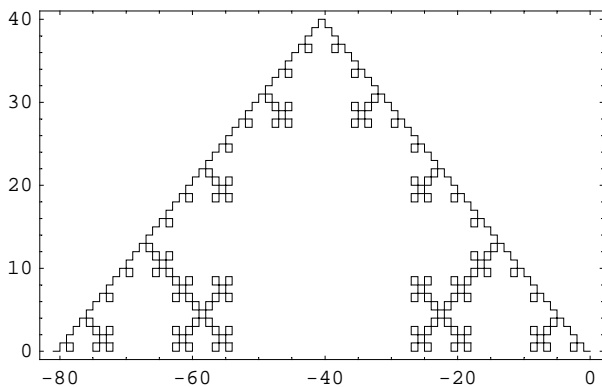
```

Um nun die Quadratic Snowflake zeichnen zu können, setzen wir für das Axiom anstelle von  $F \{Pi/2\}$  ein. für  $F$ , also für den InString setzen wir für jedes  $-F -Pi/2$  ein und für jedes  $+F Pi/2$ . Dies ergibt folgendes:

```

Clear;
Axiom = {Pi / 2};
Instr = {-Pi / 2, Pi / 2, Pi / 2, -Pi / 2};
Iterations = 4;
GoldenCut = (1 + 5 ^ (1 / 2)) / 2;
plotstring[makestring[Axiom, Instr, Iterations], 1 / GoldenCut];

```

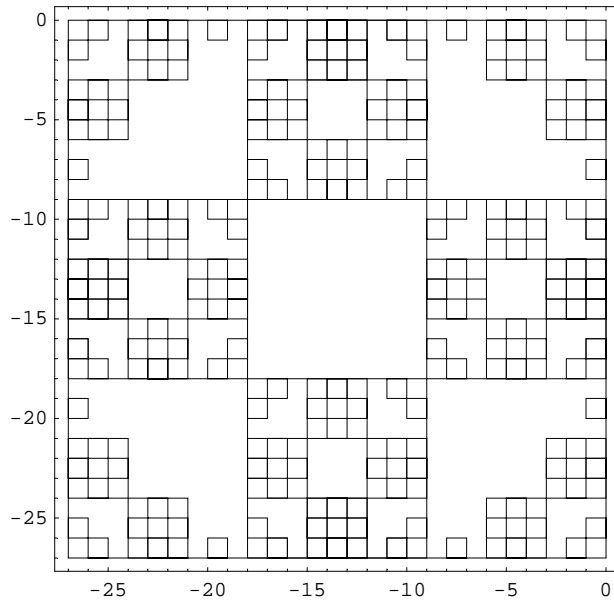


Mit diesen Hilfsfunktionen ist es nun sehr einfach, ein paar coole Figuren zu zeichnen.

---

## Board

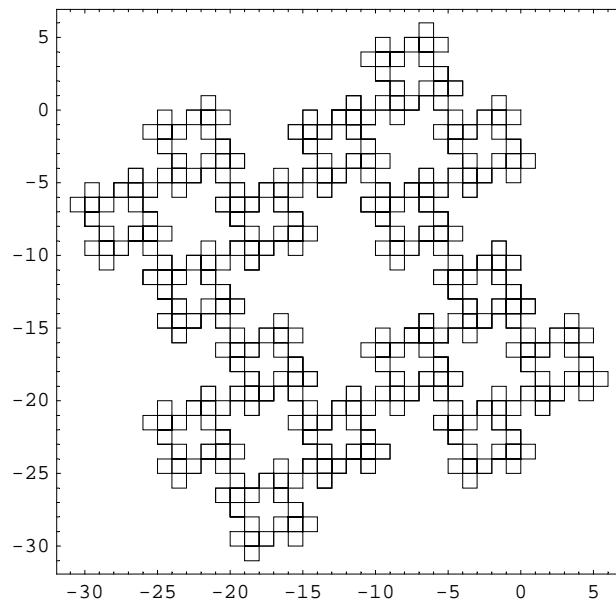
```
Clear;  
Axiom = {Pi/2, Pi/2, Pi/2, Pi/2};  
Instr = {0, Pi/2, Pi/2, Pi/2, Pi/2, 0};  
Iterations = 3;  
plotstring[makestring[Axiom, Instr, Iterations], 1];
```



---

**Cross**

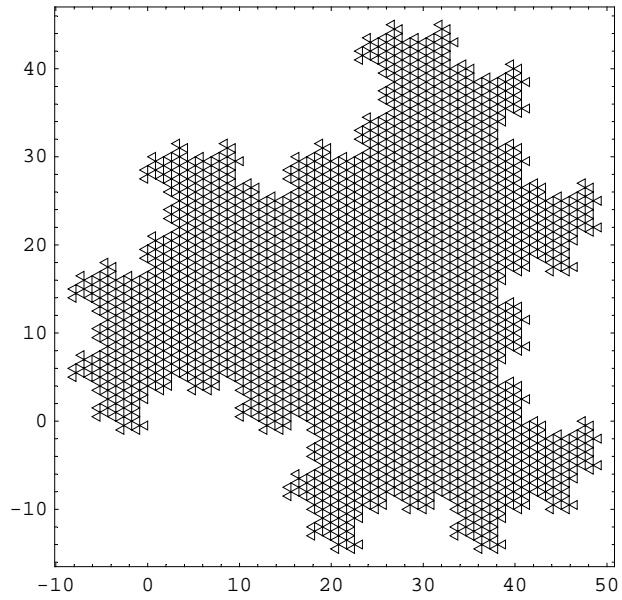
```
Clear;  
Axiom = {Pi/2, Pi/2, Pi/2, Pi/2};  
Instr = {Pi/2, -Pi/2, Pi/2, Pi/2};  
Iterations = 4;  
plotstring[makestring[Axiom, Instr, Iterations], 1];
```



---

## Triangle

```
Clear;  
Axiom = {2 * Pi / 3, 2 * Pi / 3, 2 * Pi / 3};  
Instr = {-2 * Pi / 3, 2 * Pi / 3};  
Iterations = 7;  
plotstring[makestring[Axiom, Instr, Iterations], 1];
```



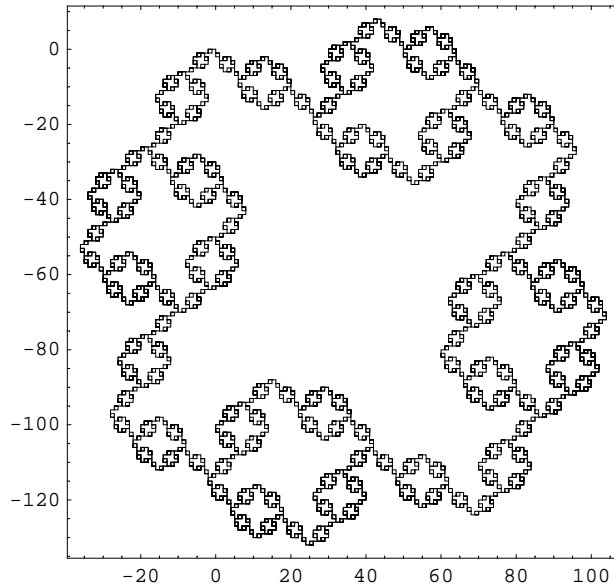
---

## Rings

```

Clear;
Axiom = {Pi / 2, Pi / 2, Pi / 2, Pi / 2};
Instr = {0, Pi / 2, Pi / 2, Pi / 2, Pi / 2, Pi / 2, -Pi / 2};
Iterations = 4;
plotstring[makestring[Axiom, Instr, Iterations], 1];

```

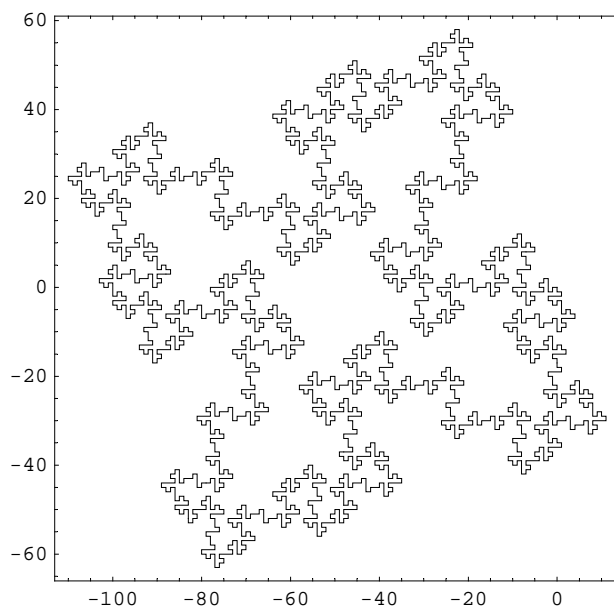


## Quadratic Koch Island

```

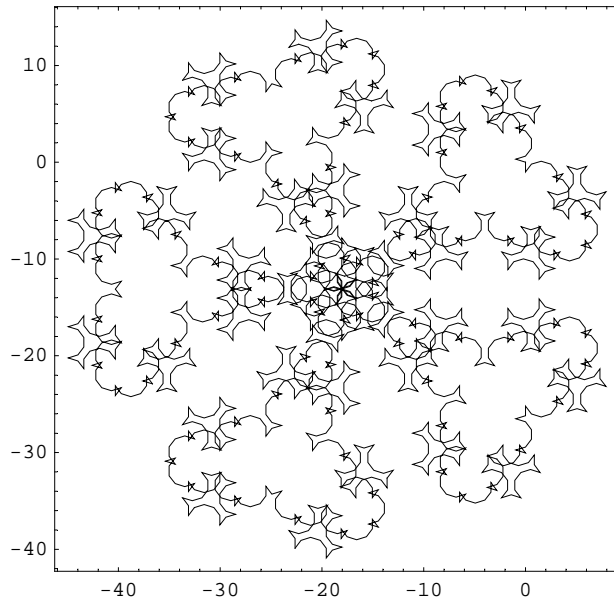
Clear;
Axiom = {Pi / 2, Pi / 2, Pi / 2, Pi / 2};
Instr = {Pi / 2, -Pi / 2, -Pi / 2, 0, 0, Pi / 2, Pi / 2, -Pi / 2};
Iterations = 3;
plotstring[makestring[Axiom, Instr, Iterations], 1];

```



## Josua + Marco 1

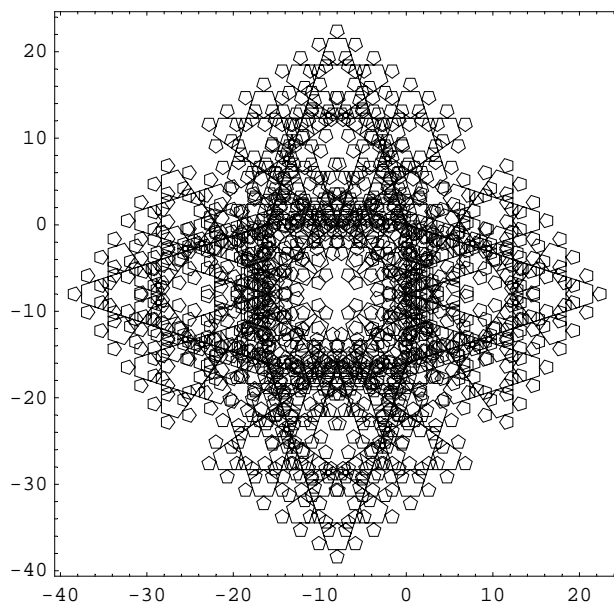
```
Clear;  
Axiom = {2*Pi/5, 2*Pi/5, 2*Pi/5, 2*Pi/5, 2*Pi/5};  
Instr = {-4*Pi/5, Pi/5, Pi/5, Pi/5, Pi/5, -4*Pi/5};  
Iterations = 3;  
plotstring[makestring[Axiom, Instr, Iterations], 1];
```



---

## Josua + Marco 2

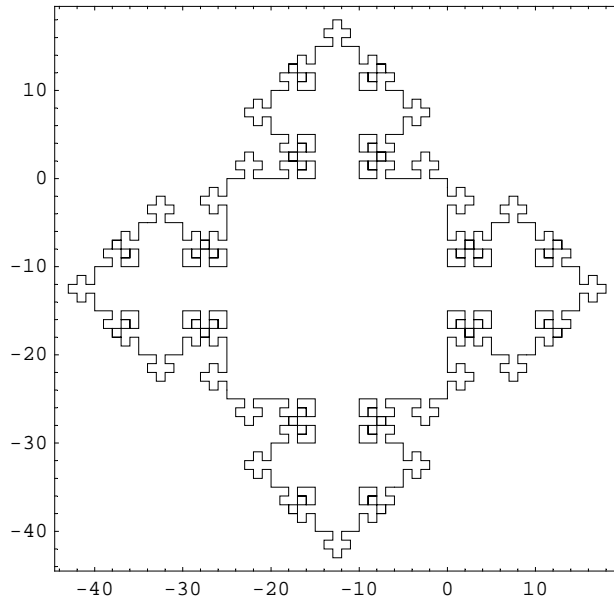
```
Clear;  
Axiom = {Pi/2, Pi/2, Pi/2, Pi/2};  
Instr = {-4*Pi/5, 2*Pi/5, 2*Pi/5, 2*Pi/5, 2*Pi/5, -4*Pi/5};  
Iterations = 4;  
plotstring[makestring[Axiom, Instr, Iterations], 1];
```



---

## Josua + Marco 3

```
Clear;  
Axiom = {Pi/2, Pi/2, Pi/2, Pi/2};  
Instr = {0, -Pi/2, -Pi/2, Pi/2, Pi/2,  
        -Pi/2, Pi/2, Pi/2, -Pi/2, Pi/2, Pi/2, -Pi/2, -Pi/2, 0};  
Iterations = 2;  
plotstring[makestring[Axiom, Instr, Iterations], 1];
```



---

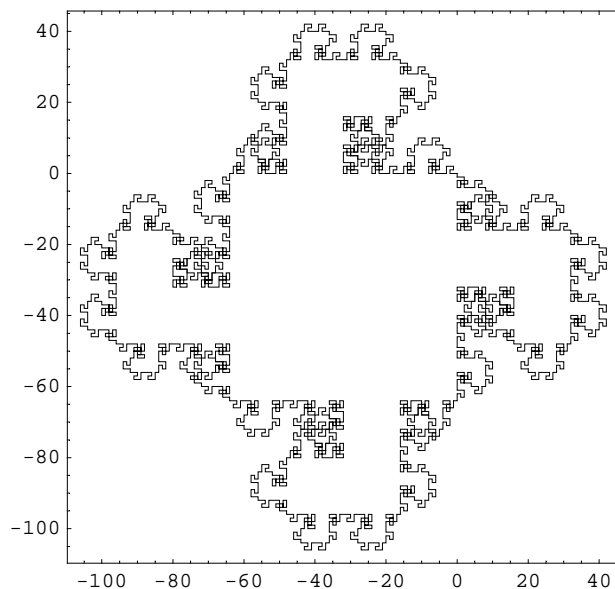
**Josua + Marco 4**



```

Clear;
Axiom = {Pi / 2, Pi / 2, Pi / 2, Pi / 2};
Instr = {0, -Pi / 2, -Pi / 2, Pi / 2, Pi / 2, 0, Pi / 2, 0, -Pi / 2};
Iterations = 3;
plotstring[makestring[Axiom, Instr, Iterations], 1];

```



Auf der erwähnten Homepage gibt es noch Figuren mit zwei InStrings (X und Y). Um auch diese Zeichnen zu können, bauten wir zuerst unsere Funktionen stringreplace und makestring so um, dass wir ihnen wirkliche Strings (anstelle von Listen) übergeben können. Danach schrieben wir noch die Funktionen stringreplacexy und makestringxy. Dafür waren nur noch kleine Anpassungen notwendig.

```

stringreplace[axiom_, instr_] := Module[{ns, i, char}, ns = "";
  For[i = 1, i <= StringLength[axiom],
    char = StringTake[axiom, {i, i}];
    If[char == "F", ns = StringJoin[ns, instr];, ns = StringJoin[ns, char];]
  ; i++]; Return[ns];

stringreplacexy[axiom_, x_, y_] := Module[{ns, i, char}, ns = "";
  For[i = 1, i <= StringLength[axiom],
    char = StringTake[axiom, {i, i}];
    If[char == "X", ns = StringJoin[ns, x];,
      If[char == "Y", ns = StringJoin[ns, y];,
        ns = StringJoin[ns, char];]
    ];
  ];
  ; i++];
  Return[ns];

makestring[Ax_, In_, It_] :=
  Module[{i, S},
    S = Ax;
    For[i = 0, i < It,
      S = stringreplace[S, In];
      i++];
    Return[S];
  ];

```

```

makestringxy[Ax_, X_, Y_, It_] :=
Module[{i, S},
  S = Ax;
  For[i = 0, i < It,
    S = stringreplacexy[S, X, Y];
    i++];
  Return[S];
];

string2table[S_, Angle_] :=
Module[{t, i, thisChar, alpha, phi}, t = {};
  phi = Angle / 360 * 2 * Pi;
  alpha = 0;
  For[i = 1, i <= StringLength[S],
    thisChar = StringTake[S, {i, i}];
    If[thisChar == "F",
      t = Join[t, {alpha}]; alpha = 0;;
    If[thisChar == "+", alpha += phi;,
      If[thisChar == "-", alpha -= phi;];
    ];
  ];
  i++;
];
Return[t];
];

plottable[S_, Ratio_] :=
Module[{i, p, f, T},
  p = {0, 0};
  T = Table[{0, 0}, {i, 1, Length[S] + 1}];
  T[[1]] = p;
  f = Pi / 2;
  For[i = 1, i <= Length[S],
    f += S[[i]];
    p = p + {Cos[f], Sin[f]};
    T[[i + 1]] = p;
    i++];
  ListPlot[T, PlotJoined -> True, Frame -> True, Axes -> False, AspectRatio -> Ratio]
];

```

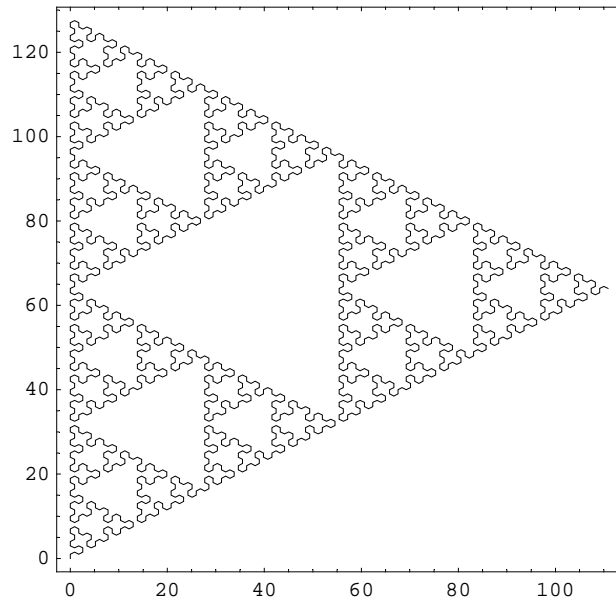
---

Nun konnten wir auch die Figuren mit zweifachen InStrings darstellen. Wir mussten uns jetzt auch nicht immer überlegen, wie das Axiom und der InString zu definieren sind, sondern konnten direkt F, X und Y verwenden.

---

## Sierpinski Arrowhead

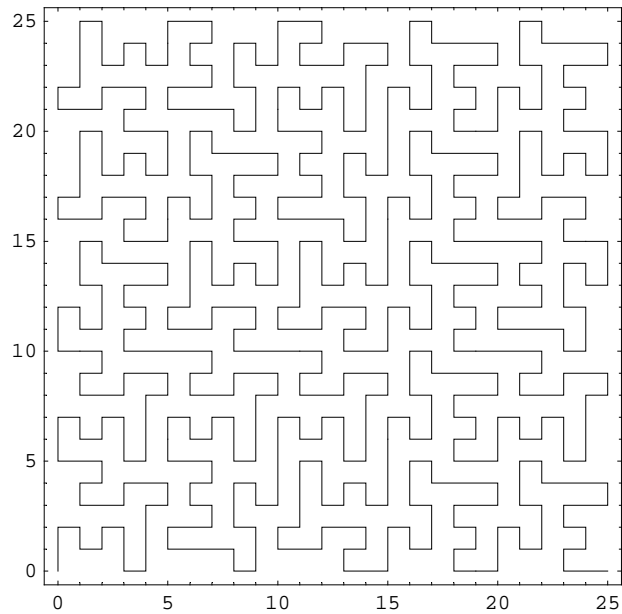
```
Clear;  
Axiom = "YF";  
Instrx = "YF+XF+Y";  
Instry = "XF-YF-X";  
angle = 60;  
Iterations = 7;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```



---

## Quadratic Gosper

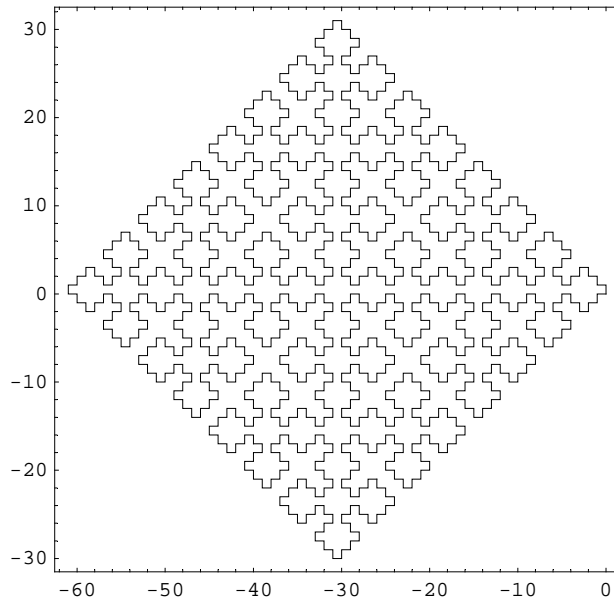
```
Clear;  
Axiom = "-YF";  
Instrx =  
  "XFX-YF-YF+FX+FX-YF-YFFX+YF+FXFXFYF-FX+YF+FXFX+YF-FXYF-YF-FX+FX+YFYF-";  
Instry = "+FXFX-YF-YF+FX+FXFYF+FX-YFYF-FX-YF+FXFYFYF-FX-YFFX+FX+YF-YF-FX+FX+YFY";  
angle = 90;  
Iterations = 2;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```



---

## Square Sierpinski

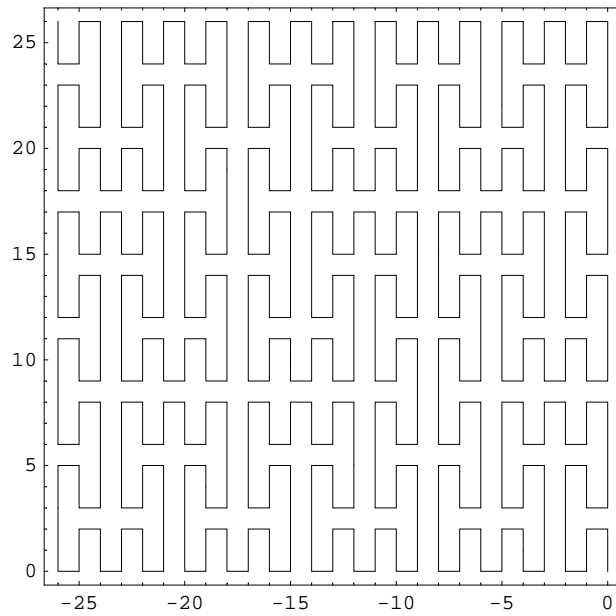
```
Clear;  
Axiom = "F+XF+F+XF";  
Instrx = "XF-F+F-XF+F+XF-F+F-X";  
Instry = "";  
angle = 90;  
Iterations = 4;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```



---

## Peano Curve

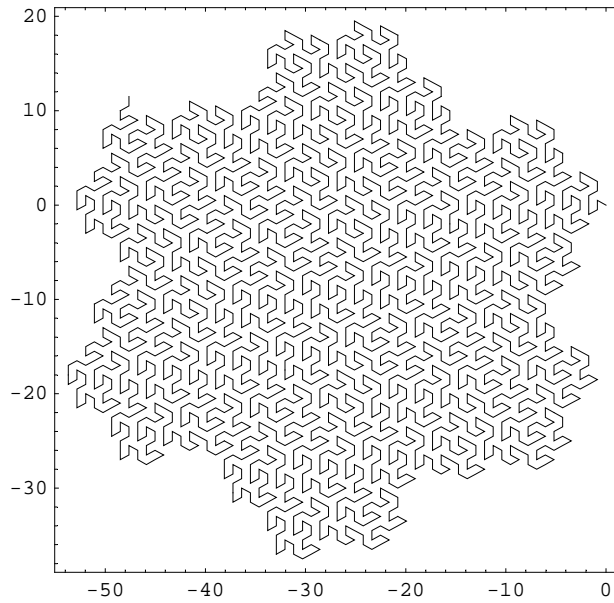
```
Clear;  
Axiom = "X";  
Instrx = "XFYFX+F+YFXFY-F-XFYFX";  
Instry = "YFXFY-F-XFYFX+F+YFXFY";  
angle = 90;  
Iterations = 3;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```



---

## Hexagonal Gosper

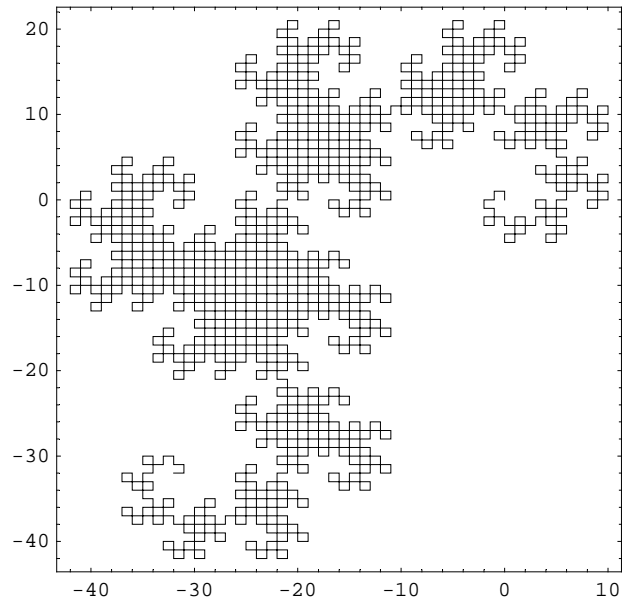
```
Clear;  
Axiom = "XF";  
Instrx = "X+YF++YF-FX--FXFX-YF+";  
Instry = "-FX+YFYF++YF+FX--FX-Y";  
angle = 60;  
Iterations = 4;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```



---

## Dragon Curve

```
Clear;  
Axiom = "FX";  
Instrx = "X+YF+";  
Instry = "-FX-Y";  
angle = 90;  
Iterations = 11;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```

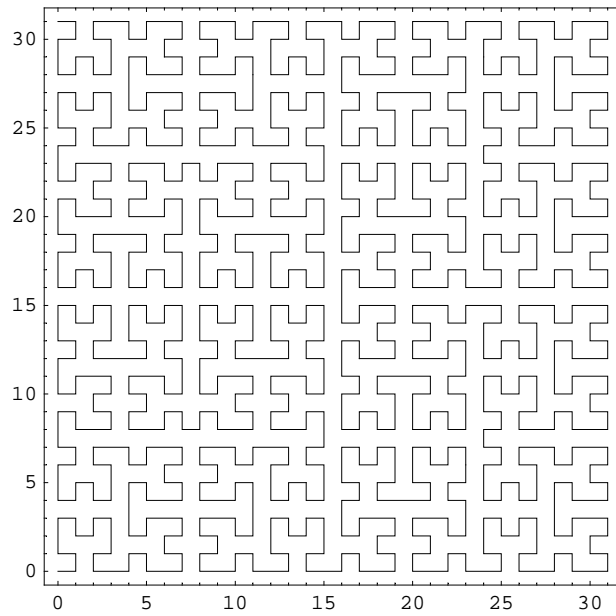


---

## Hilbert



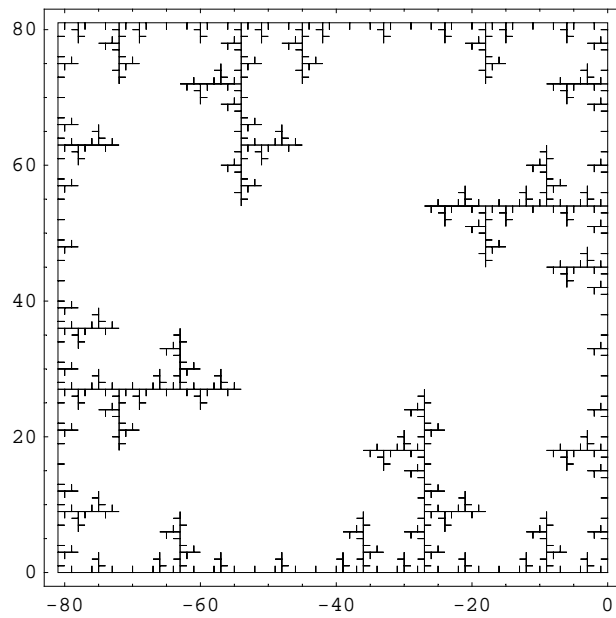
```
Clear;  
Axiom = "X";  
Instrx = "-YF+XFX+FY-";  
Instry = "+XF-YFY-FX+";  
angle = 90;  
Iterations = 5;  
plottable[string2table[makestringxy[Axiom, Instrx, Instry, Iterations], angle], 1];
```



---

## Crystal

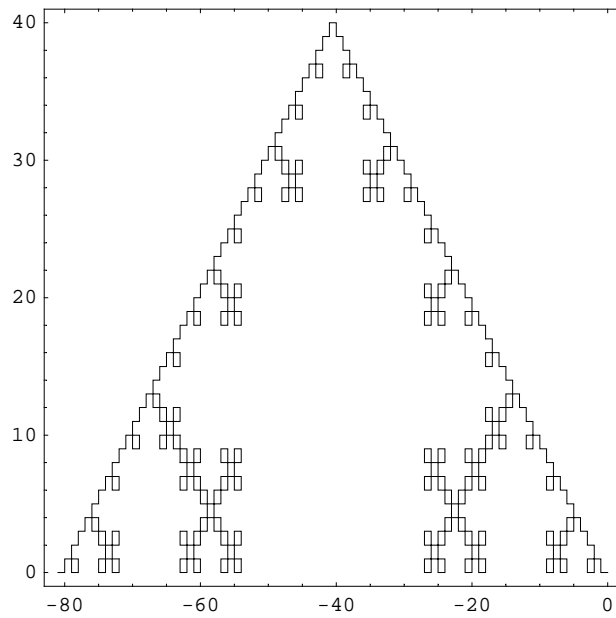
```
Clear;  
Axiom = "F+F+F+F";  
Instr = "FF+F++F+F";  
angle = 90;  
Iterations = 4;  
plottable[string2table[makestring[Axiom, Instr, Iterations], angle], 1];
```



---

## Quadratic Snowflake

```
Clear;  
Axiom = "+F";  
Instr = "F-F+F+F-F";  
angle = 90;  
Iterations = 4;  
plottable[string2table[makestring[Axiom, Instr, Iterations], angle], 1];
```



---

## Triangle

```
Clear;  
Axiom = "F+F+F";  
Instr = "F-F+F";  
angle = 120;  
Iterations = 5;  
plottable[string2table[makestring[Axiom, Instr, Iterations], angle], 1];
```

