Gauss Plane and Complex Function

Graphics primitives

```
In[1]:= b[x__] := Map[{Re[#], Im[#]} &, x, {2}]

In[2]:= bl[x__] := Map[Line, b[x]]

In[3]:= c[x__] := Map[{Re[#], Im[#]} &, Transpose[x], {2}]

In[4]:= cl[x__] := Map[Line, c[x]]
```

Set of Complex Numbers (Lattice on Gauss plane)

```
In[5]:= a = Table[N[i / 10] + I N[j / 10], (*I=Sqrt[-1]*)
          {j, 0.1, 101, 10.1 / 2}, {i, 0.1, 101, 10.1 / 2}];
```

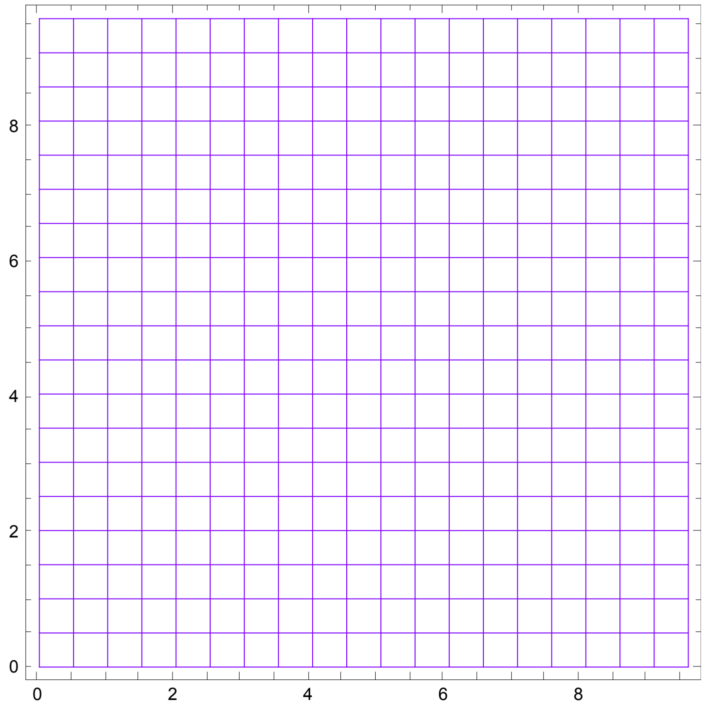
Function, shows lattice

```
In[6]:= plt[clr_, x__] := Show[Graphics[{Hue[clr], bl[x]}],
         Graphics[{Hue[clr], cl[x]}], AspectRatio → Automatic, Frame → True]
```

Show Lattice.

In[7]:=
```
plt[.75, a]
```

Out[7]=

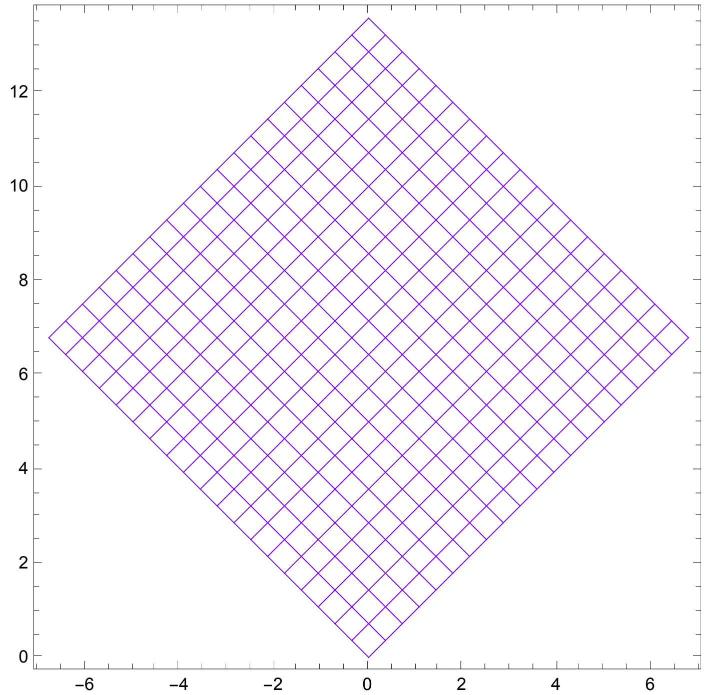


In[8]:=
```
r = Exp[I Pi * .25]     (*I=Sqrt[-1]*)
```

Out[8]= 0.707107 + 0.707107 ⅈ

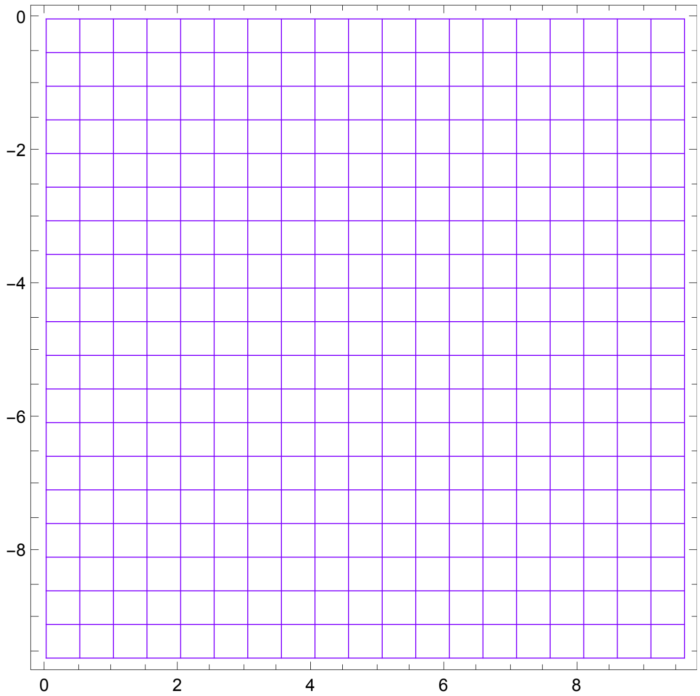
In[9]:=
```
plt[.75, r * a]
(*Rotate of Angle=Pi/4 around zero.r acts to each element of a*)
```

Out[9]=



In[10]:=
```
plt[.75, Conjugate[a]]
(*Conjugate acts to each element of a*)
```

Out[10]=

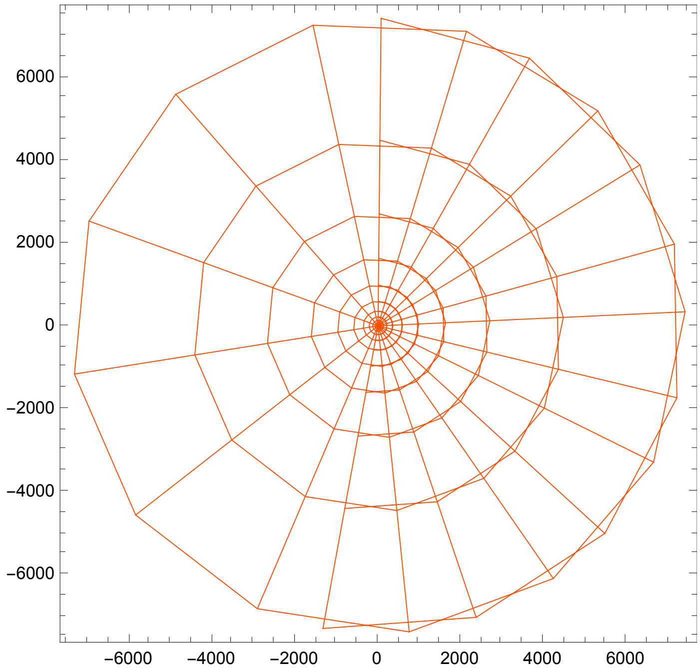


Mapping Lattice

Function acts to each element of List.

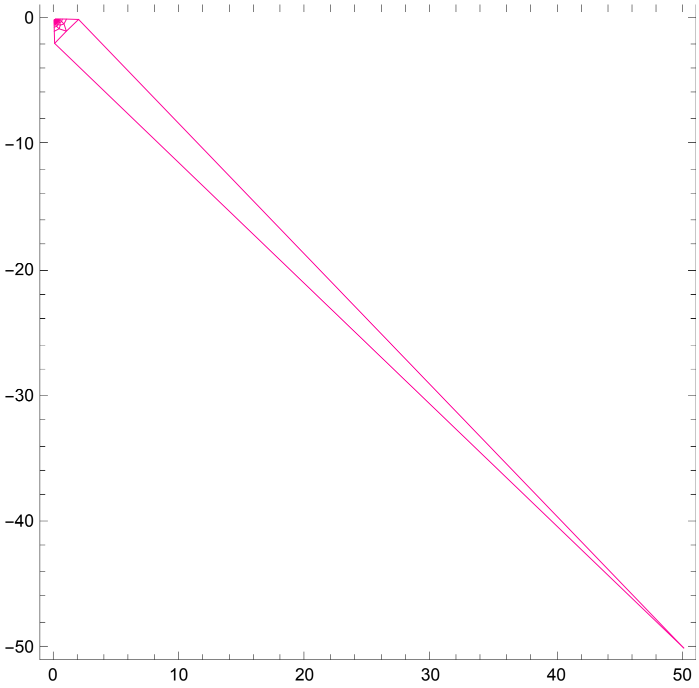
In[11]:=
```
plt[.05, Sin[a]]     (*sine function*)
```

Out[11]=



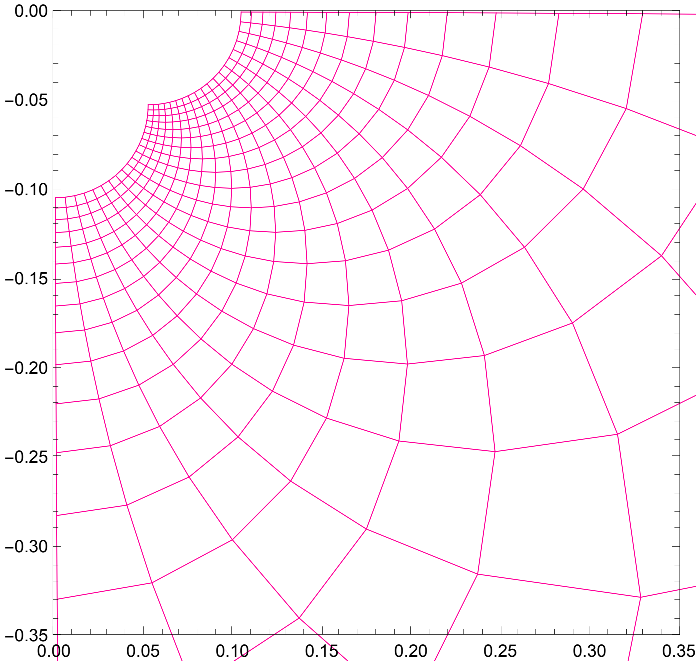
In[23]:=
```
g1 = plt[.9, 1 / a] (*1/a is inverse of each element of a*)
```

Out[23]=



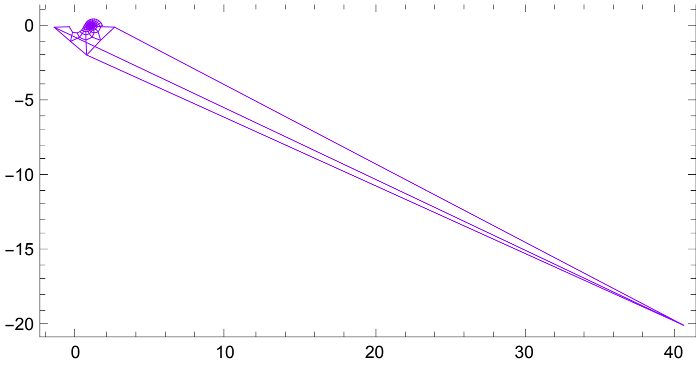
In[24]:=
```
Show[g1, PlotRange → {{0, 0.35}, {-0.35, 0}}]
```

Out[24]=



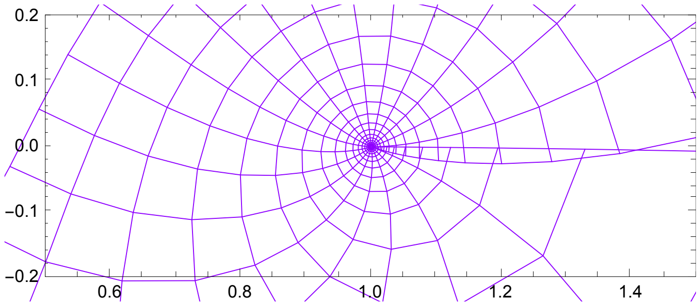
In[25]:=
```
g2 = plt[.76, Zeta[a]]      (*Riemann's Zeta function*)
```

Out[25]=



In[27]:=
```
Show[g2, PlotRange → {{0.5, 1.5}, {-0.2, 0.2}}]
```

Out[27]=



In[15]:=
```
? BesselJ
```

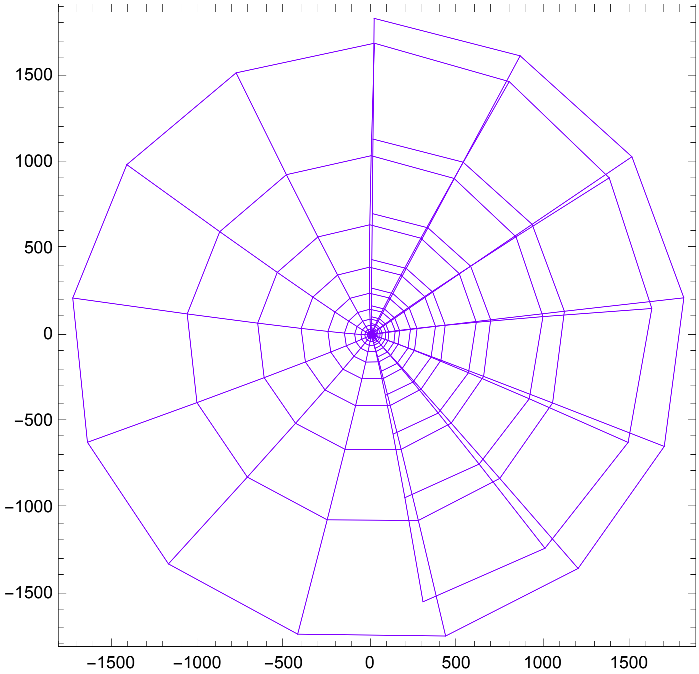
Out[15]=

Symbol

BesselJ[n, z] gives the Bessel function of the first kind $J_n(z)$.

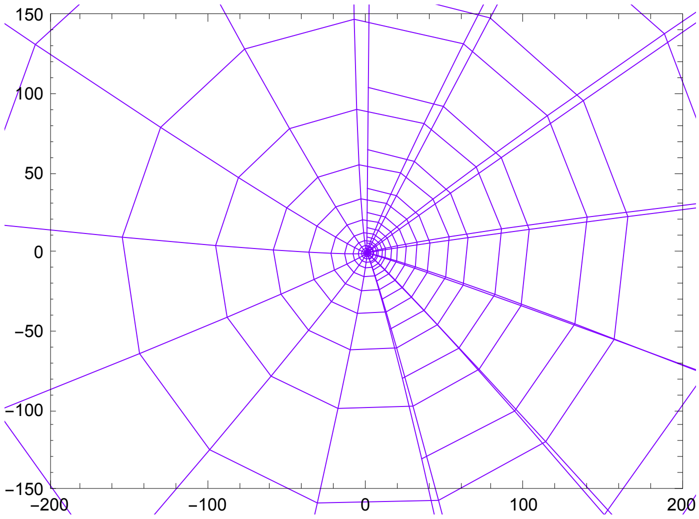
In[28]:=
```
g3 = plt[.75, BesselJ[1, a]]
```

Out[28]=



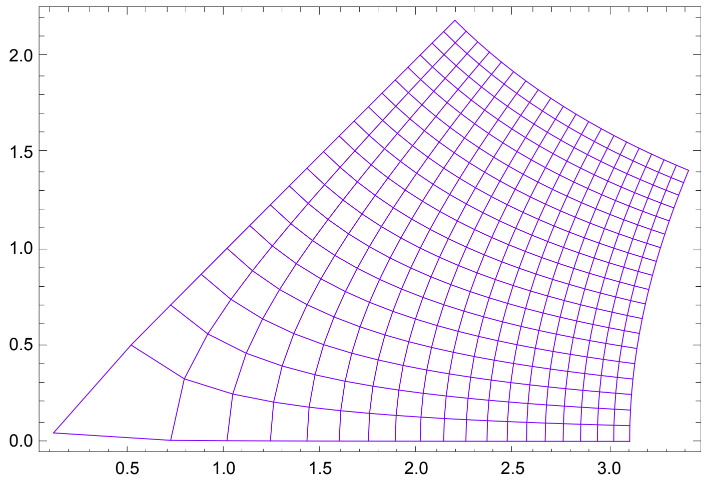
In[29]:=
```
Show[g3, PlotRange → {{-200, 200}, {-0150, 150}}]
```

Out[29]=



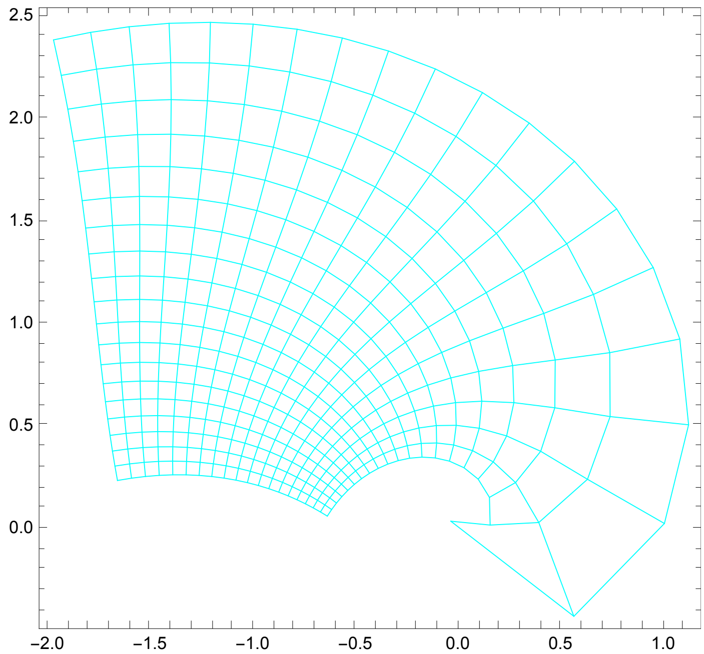
In[16]:=
```
plt[.75, Sqrt[a]]
(*Square Root of each element of a*)
```

Out[16]=



In[17]:=
```
plt[.5, a^(.5+I)]     (*Power by "1/2 + Sqrt[-1]"*)
```

Out[17]=



In[18]:=
```
?AiryAi
```

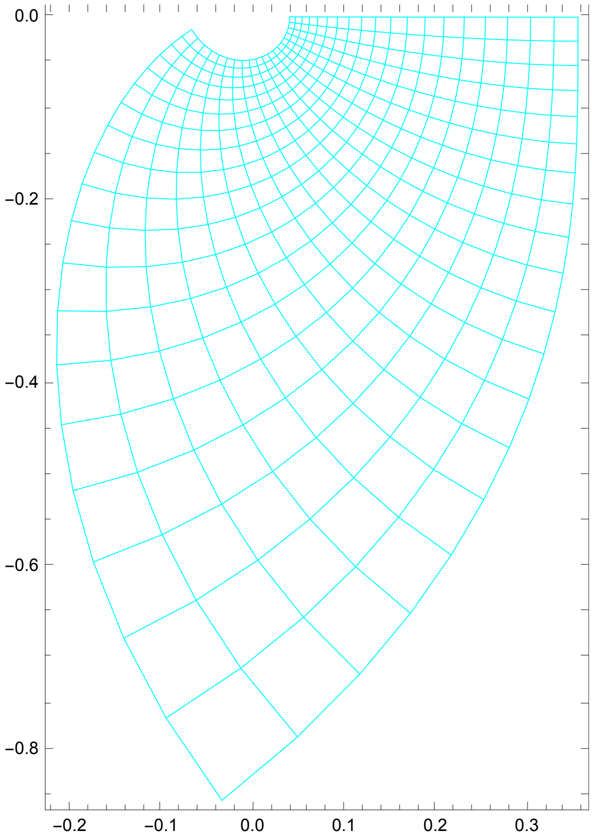
Out[18]=

Symbol

AiryAi[z] gives the Airy function Ai(z).

In[19]:=
```
plt[.5, AiryAi[Evaluate[a / 5]]]
```

Out[19]=



In[20]:=
```
a1 = 4 + 4 I
```

Out[20]=

4 + 4 ⅈ

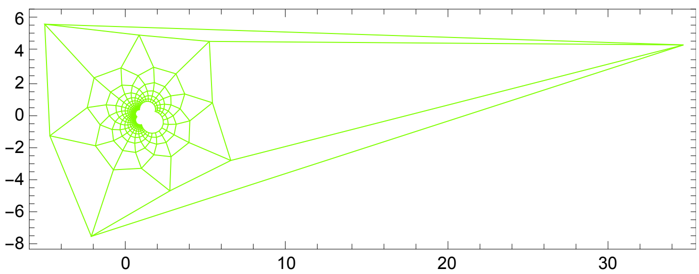
In[21]:=
```
a2 = 6 + 7 I
```

Out[21]=

6 + 7 ⅈ

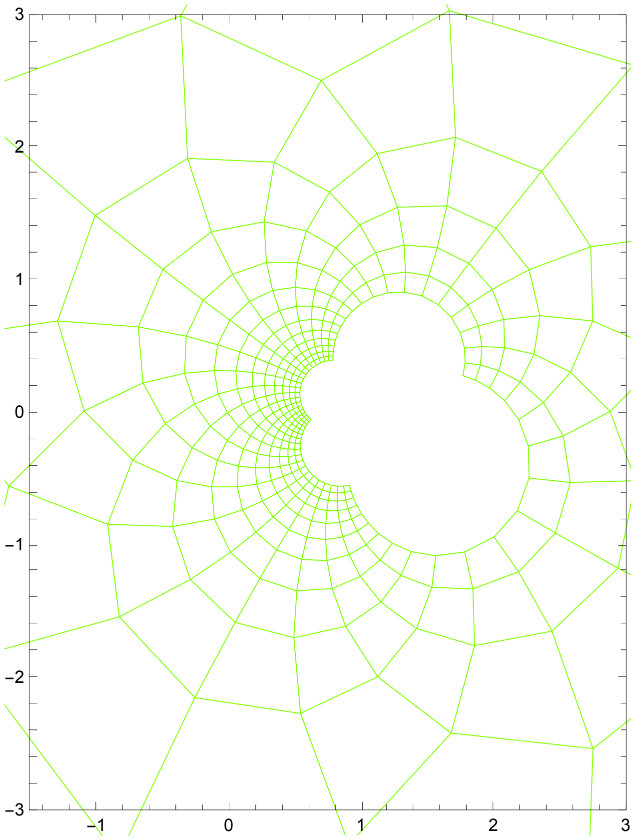
In[30]:=
```
g4 = plt[.25, (a - a1) / (a - a2)]
```

Out[30]=



In[31]:=
```
Show[g4, PlotRange → {{-1.5, 3}, {-3, 3}}]
```

Out[31]=



Biblography

Programming in Mathematica Second Edition, R . Maeder, Addison - Wesley, 1991