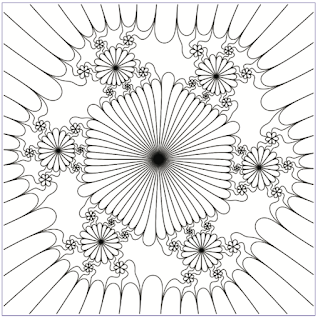
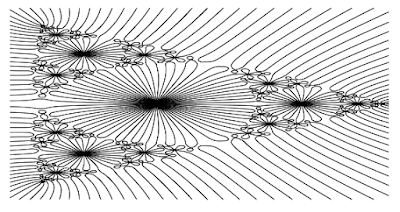
I was recently fascinated by the illustration taken from the leaflet <https://www.maa.org/sites/default/files/pdf/Mathhorizons/pdfs/ColoringPage_MH_Nov17.pdf>



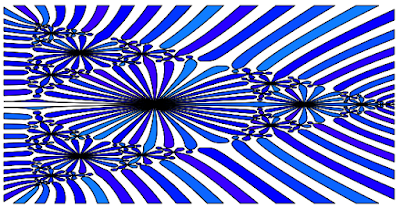
I knew that complex functions could be colored in mesmerizing ways but the little book by Julie Barnes, William Kreahling, and Beth Schaubroeck, “Coloring book of complex function repesentation”, MAA Press, 2017, explores new avenues and, say, I never realized that contour lines could be used to obtain beautiful images. Moreover, I teach students how to draw contour curves of two dimensional functions like \(f(x,y)\) and thought this could be a funny example (though, it is really another story when you tweak functions of the complex plane in itself.) Hence, I bought the book (actually, I got it out of the bonus points awarded by writing reviews on Mathscinet) and decided to give R a try. Take the function \[ f(x)=z^2-\frac{0.6}z, \] where \(z\in\mathbb{C}\) and consider 6 nested applications of the function with itself \[ f^6(z)=f(f(f(f(f(f(z)))))):=f6(z). \] (don’t ask me how the authors found that, the book provides plenty of insights and I do not want to spoil all the fun!) Of course, \(f6(z)\) is a complex number and we are going to plot the \(z\)’s such that the real part of \(f6(z)\) is null. In other words, we plot the zero-contour level of \(\Re (f6)\) on the rectangle \([-0.85,1.35\times [-1.4,1.4]\) of the complex plane. If you prefer, we want to find the complex points that are mapped to the imaginary axis after 6 applications of \(f\). The standard (so to say!) way to plot contour lines in is to use outer and contour. This time, I’m not plotting \(f(x,y)\) where \(x,y\) are two real numbers but I’ll compute \(f6(x+i\ y)\), where \(z=x+i\ y\):

fz <- function(z) z^2-0.6/z  
fz6 <-function(z) fz(fz(fz(fz(fz(fz(z))))))  
f6 <- function(x,y) fz6(x+1i\*y)  
x <- seq(-0.85,1.35,len=1001)  
y <- seq(-1.4,1.4,len=1001)  
m <- outer(x,y,f6) # this is a matrix of complex numbers  
contour(x,y,Re(m),levels=c(0),drawlabels=F,axes=F)   
# Re(m) takes the real parts



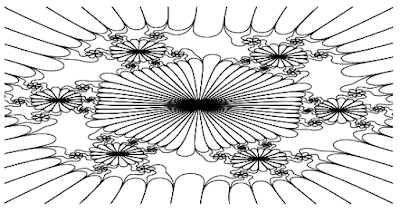
I was pretty much happy of the work: a few lines of code fill the matrix \(m\) with all the complex values of \(f6\) for all \(x+i\ y\) (\(x\) and \(y\) are taken from the sequences). Then, a contour on the real part of the matrix is displaying this marvel, if you share my enthusiasm… I decided to tamper a bit with the filling of the contours using colors and, in so doing, showing how to extract the points contained in all in all the (possibly many) components of contour lines:

cl <- contourLines(x,y,Re(m),levels=0)  
pal <- topo.colors(48)  
contour(x,y,Re(m),levels=c(0),drawlabels=F,axes=F)  
for(i in 1:length(cl)) polygon(cl[[i]]$x,cl[[i]]$y,col=pal[i %% 12+1])



The variable cl is a list with the points of all the components of the zero-level contours and pal is a palette of colors (you can unleash your imagination and change palette). Then a for loop fills with different shades every single contour in I decided to use this and other variations to send Christmas cards to friends and relatives. I love math and (also) wrote the complex function \(f(z)\) on the card but really felt the pictures were nice and needed no complexity (!) at all to be enjoyed. The picture that originates all the story is related to another complex function and displays three contours at one time. Easy as a piece of cake:

fz <- function(z) z^3+(-0.2+0.11\*1i)/z^3  
fz3 <-function(z) fz(fz(fz(z)))  
f3 <- function(x,y) fz3(x+1i\*y)  
x <- seq(-1.08,1.08,len=1001)  
y <- seq(-1.3,1.3,len=1001)  
m <- outer(x,y,f3) # this is a matrix of complex numbers  
contour(x,y,Re(m),levels=c(-2,0,2),drawlabels=F,axes=F)   
# Re(m) takes the real parts



Have a colorful (and not that complex) new yeaR!