import Graphics.UI.GLUT

myPoints :: [(GLfloat,GLfloat,GLfloat)]

myPoints = [ (sin (2\*pi\*k/12), cos (2\*pi\*k/12), 0) | k <- [1..12] ]

main :: IO ()

main = do

(\_progName, \_args) <- getArgsAndInitialize

\_window <- createWindow "Hello World"

displayCallback $= display

mainLoop

display :: DisplayCallback

display = do

let color3f r g b = color $ Color3 r g (b :: GLfloat)

vertex3f x y z = vertex $ Vertex3 x y (z :: GLfloat)

clear [ColorBuffer]

renderPrimitive Quads $ do

color3f 1 0 0

vertex3f 0.1 0.1 1

vertex3f 0 0.2 1

vertex3f 0.1 0.3 1

color3f 0 0 0

vertex3f 0.1 0.1 1

vertex3f 0.3 0.1 1

vertex3f 0.1 0.3 1

vertex3f 0.3 0.3 1

color3f 1 0 1

vertex3f 0.2 0 1

vertex3f 0.1 0.1 1

vertex3f 0.3 0.1 1

color3f 1 0 1

vertex3f 0.2 0 1

vertex3f 0.1 0.1 1

vertex3f 0.3 0.1 1

color3f 1 1 1

vertex3f 0.3 0.1 1

vertex3f 0.4 0.2 1

vertex3f 0.3 0.3 1

color3f 0 1 1

vertex3f 0.1 0.3 1

vertex3f 0.3 0.3 1

vertex3f 0.2 0.4 1

color3f 0 1 1

vertex3f 0.1 0.3 1

vertex3f 0.3 0.3 1

vertex3f 0.2 0.4 1

flush

