PROJECT #C: TRACING IMPLICIT CURVES ACROSS A QUADTREE

ABEL GOMES

This project aims at developing in students the sense of mastering computational geometry techniques together with numerical methods, being here these methods used for sampling implicit curves in 2D.

Project #C. Let us design the program to trace a regular implicit curve in a domain $D \subset \mathbb{R}^2$. This domain D is sometimes called bounding box and must be subdivided using a quadtree or any other axis-aligned subdivision scheme in order to sample the curve that lies in D. A number of functions have to be designed and implemented in C, including the following:

(1) A function that describes a curve (for example, a straight line $y - x - 1 = 0$):

```
float f(float x,float y) { return y-x-1;}
```

(2) A function that returns the discrete derivative (or forward finite differences) along x-direction:

```
float dX(float x, float y)  return (f(x+h,y)-f(x,y))/h;
```

where h is an infinitesimal given, for example, by 10^{-7}.

(3) A function that returns the discrete derivative along y-direction:

```
float dY(float x, float y)  return (f(x,y+h)-f(x,y))/h;
```

(4) A function that returns the x-value of a curve point in a line that is parallel to x-axis. It implements the Newton iterator in the x-direction:

```
float NewtonX (float x, float y)
```

The arguments x and y constitute the pair of the initial estimate or point. Besides, the argument y is constant and gives us a line parallel to x-axis. This function returns the final value of x. (This was explained in two consecutive classes, so have

---

Date: Assigned: December 13, 2013; Due: January 20, 2014; Visual Computing and Multimedia, Fall 2013-2014.

a look on your lecture notes.)

(5) A function that returns the y-value of a curve point in a line that is parallel to y-axis. It implements the Newton iterator in the y-direction:

```
float NewtonY (float x, float y)
```

(This was explained in two consecutive classes, so have a look on your lecture notes.)

(6) A function to trace implicit curves in the bounding box $D = [x_{min}, x_{max}] \times [y_{min}, y_{max}] \subset \mathbb{R}^2$:

```
void 2DImplicitCurve(float xmin, float xmax, float ymin, float ymax,
float width)
```

where `width` is the line width used to draw the curve.

The OpenGL main program to generate implicit curves on screen is then as follows:

```
  int main (int argc, char * argv[])
{
  glutInit(&argc,argv);
  glutInitDisplayMode(GLUT_RGB | GLUT_DOUBLE);
  glutInitWindowSize(Xwidth,Yheight);
  glutCreateWindow("Tracing Implicit Curves in the Plane");
  glutDisplayFunc(2DImplicitCurve);
  glutKeyboardFunc(myKeyboard);
  glutMainLoop();
  return 0;
{
```