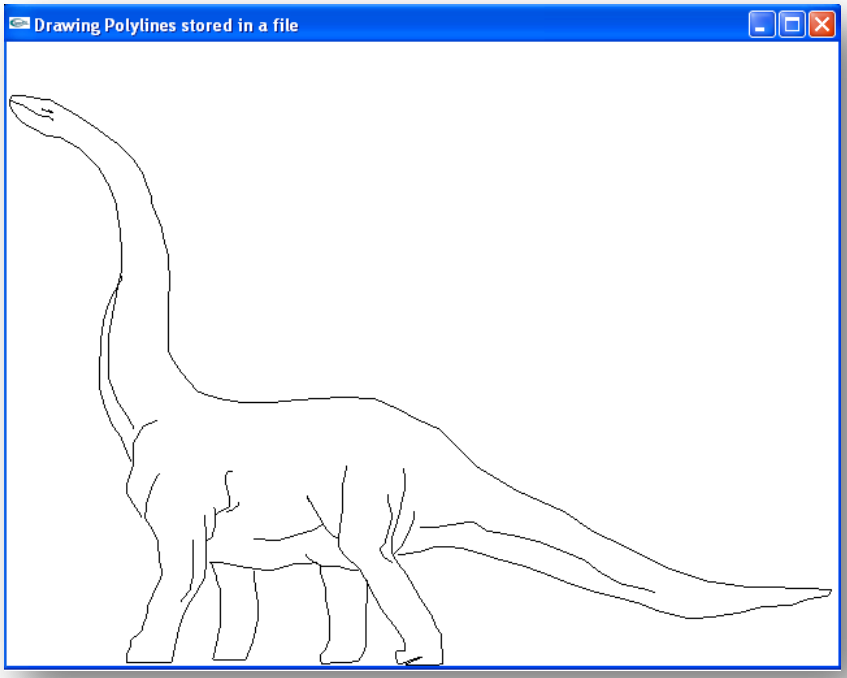
AL GHURAIR UNIVERSITY
College of Computing

CSC 305 **Computer Graphics**

Lab Exercise

Drawing polylines that have been stored in a file. Most interesting pictures made up of polylines contain a large number of line segments. In such cases it's important to store a description of the polylines in a file, so that the picture can be redrawn at will.

Sample Output



Source Code

```
#include "stdafx.h"
#include <fstream>

#include <GL/glut.h>


void setWindow(float left, float right, float bottom, float top)
{
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(left, right, bottom, top);
}

void drawPolyLineFile(char * fileName)
{
	std::ifstream inStream;
	inStream.open(fileName); // open the file
	if(inStream.fail())
		return;
	//glClear(GL_COLOR_BUFFER_BIT);	// clear the screen
	GLint numpolys, numLines, x ,y;
	inStream >> numpolys; // read the number of polylines
	for(int j = 0; j < numpolys; j++) // read each polyline
	{
		inStream >> numLines;
		glBegin(GL_LINE_STRIP); // draw the next polyline
		for (int i = 0; i < numLines; i++)
		{
			inStream >> x >> y; // read the next x, y pair
			glVertex2i(x, y);
		}
		glEnd();
	}
	glFlush();
	inStream.close();
}

//<<<<<<<<<<<<<<<<<<<<<<<< myInit >>>>>>>>>>>>>>>>>>>>
void myInit(void)
{
	glClearColor(1.0,1.0,1.0,0.0); // set white background color
	glColor3f(0.0f, 0.0f, 0.0f); // set the drawing color
	glPointSize(4.0); // a 'dot' is 4 by 4 pixels

	setWindow(0.0,640.0, 0.0, 480.0);

}

//<<<<<<<<<<<<<<<<<<<<<<<<< myDisplay >>>>>>>>>>>>>>>>>

void myDisplay(void)
{
	glClear(GL_COLOR_BUFFER_BIT); // clear the screen

	drawPolyLineFile("dino.dat");

	glFlush(); // send all output to display
}
```

```
//<<<<<<<<<<<<<<<<<<<<<<<<< main >>>>>>>>>>>>>>>>>>>>>>>

void main(int argc, char** argv)
{
  glutInit(&argc, argv); // initialize the toolkit
  glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB); // set display mode
  glutInitWindowSize(640,480); // set window size
  glutInitWindowPosition(100, 150); // set window position on screen
  glutCreateWindow("Drawing Polylines stored in a file"); // open the screen window
  glutDisplayFunc(myDisplay); // register redraw function
  myInit();
  glutMainLoop(); // go into a perpetual loop
}
```