

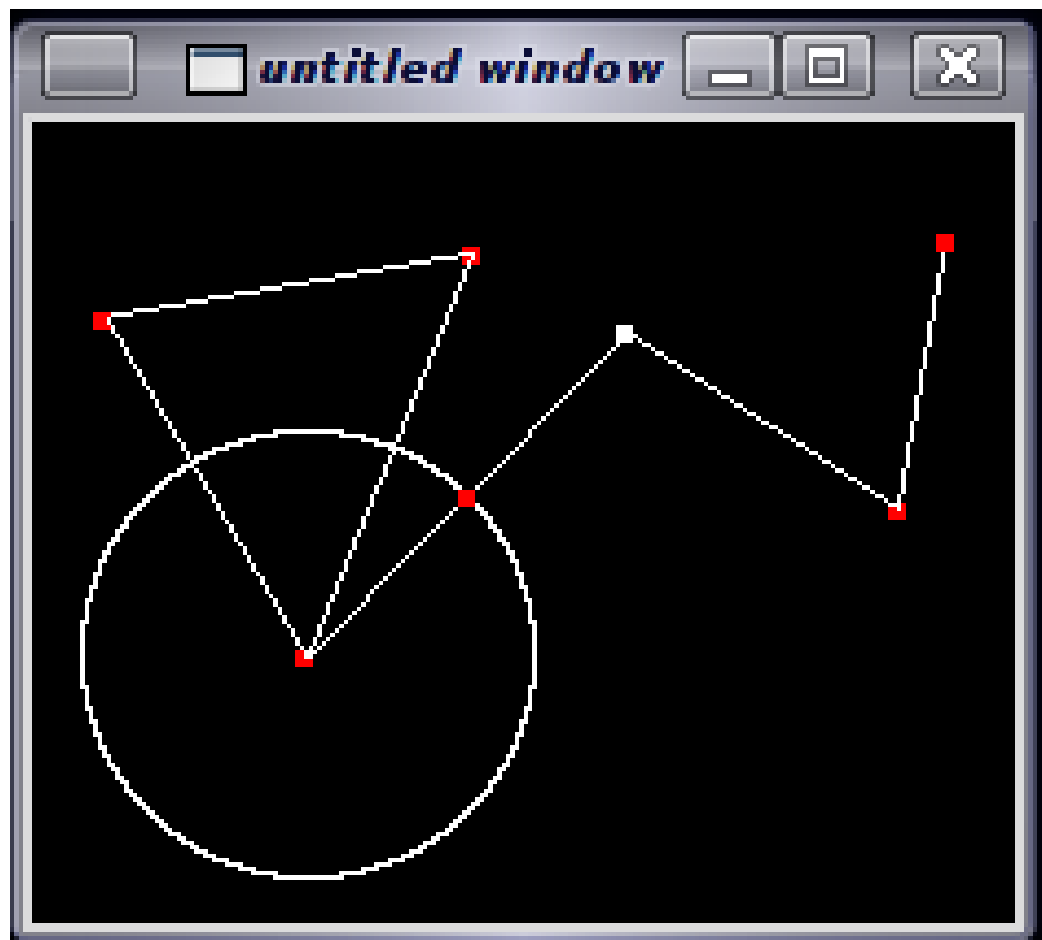
Interactive Geometry in 3D

Jacob Welsh

TJHSST Computer Systems Lab 2007-2008

Abstract

The goal of this project is to write a program that allows its user to create and manipulate a complex system of geometric objects in space. From a few basic object types, interesting and useful constructions can be built. This could be useful for education, mathematical or scientific research or visualization, or just for fun.



A screenshot showing points, line segments, and a circle, created by clicking in various locations on the screen in different drawing modes. The points can be selected and dragged with the mouse, and the other objects are updated accordingly.

Results

Many of the desired features were implemented, such as drawing lines, points and circles, and selecting points and dragging them, with the dependent objects moving accordingly. The code supports three dimensions, but this does not currently work very well due to difficulties with projecting screen to world coordinates.

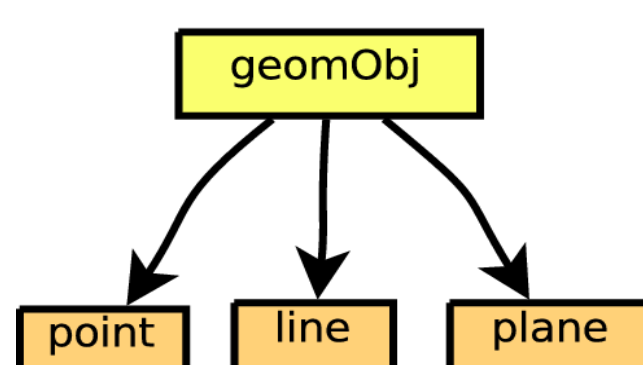
The program was implemented in C, using SDL and OpenGL for graphics. Geometric objects were represented as a tree of structures, using pointers to achieve a form of inheritance for the specific types of objects.

Background

For a while there has been software for computer assisted design (CAD), which utilizes a few basic shapes and techniques such as snapping and numeric entry to create precise, polished diagrams of a product that can then be used in its manufacturing.

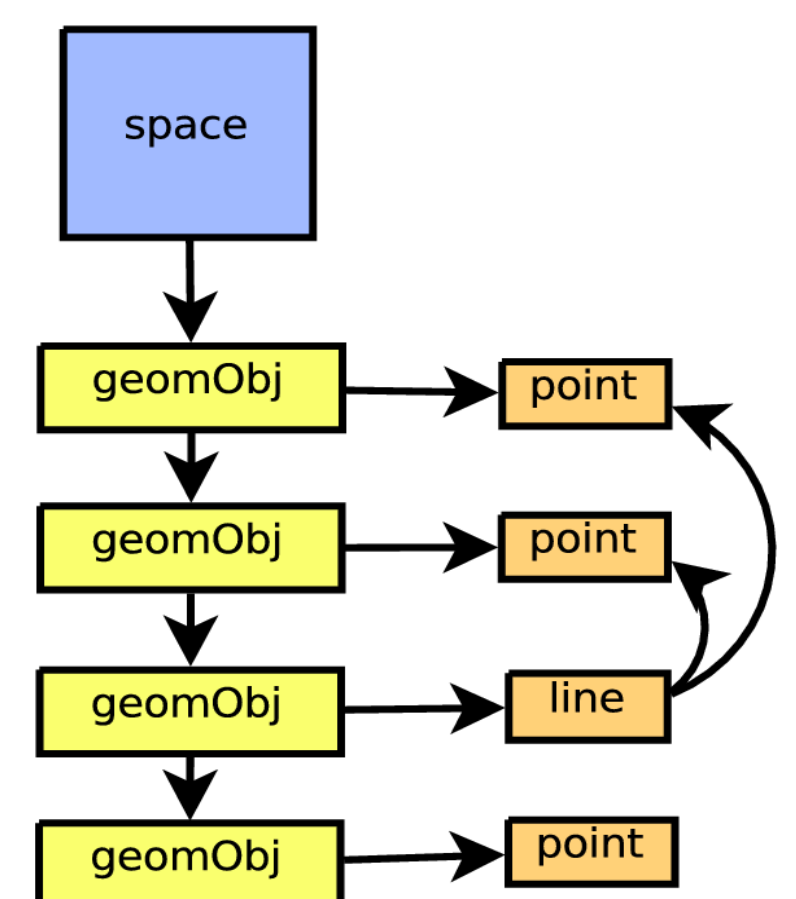
A similar sort of program is used for 3D modeling, in which the user constructs polygon meshes in three dimensions: freehand; with snapping; and numerically. My program aims to be more focused on geometric objects and dynamic preservation of their relationships as some are manipulated. The leading example of this is a commercial program called The Geometer's Sketchpad. Its interface is heavily mouse-based and rather inefficient, and it is limited to two dimensions. However, the fact that it is possible to build primitive pseudo-3D constructions in it illustrates the power behind the idea of geometric construction. The basic philosophy for the user interface of my program comes from the modeling program Blender and the text editor Vi.

Object Hierarchy



"Object Oriented"
- Inheritance
- Built with structs and pointers

Global Linked List for Drawing



A schematic diagram showing the internal structure of geometric objects, and the structure that stores them for display and calculation purposes.