always declare your variables with GL-defined types, such as

\texttt{GLuint, GLsizei, GLdouble}

for a length \& vector, i.e., C array of \texttt{GLint}.

\texttt{GLfloat*} (for a length \& vector, i.e., \texttt{C} array of \texttt{GLfloat})

different kinds of arguments. E.g., \texttt{GLfloat2}, \texttt{GLfloat3}, \texttt{GLfloat4},

like \texttt{GLfloat} to describe a family of functions, each taking

numbers and types of the arguments. Documentation needs names

suffix numbers and letters of OpenGL function names indicate

but OpenGL accepts integer data also.

- OpenGL TIPS: Floating point (C/C++ float or \texttt{double}) is used,

- useful range of world coordinates.

- Important opportunity: \texttt{X} you choose units, orientation, meaning, and

pictures of it.

model. It is better to separate world modeling from creating 2-d

\texttt{World coordinates: Use to define places within the whole "world"}

Various coordinate systems and what they are used for:
The "world model" 

The master will be copied (after translation) into one or more places in the template or master copy of a more or less complex graphical object.

Modeling (or local or master) coordinates: Used to define a
double.c

This is a simple double buffered program.

Pressing the left mouse button rotates the rectangle.

Pressing the middle mouse button stops the rotation.

Copyright (c) 1993-1997, Silicon Graphics, Inc.

ALL RIGHTS RESERVED

legalense deleted.

DOWNLOAD from http://www.opengl.org

EG what happens if double buffering is NOT USED? Try it and see!

Prepare for future projects: Read about the animation of pages 20-26 of the RedBook. Download, play with, and MODIFY the double.c code.
Towards the negative z axis, therefore, just use x and y coordinates to OpenGL default: default OpenGL viewing is FROM the positive z axis.

重度特效 CHANGES the current (modelview) transformation matrix.

getPushMatrix() saves a copy of the current (modelview) transformation.

setTexture(GL::COLOR_BUFFER_BIT);

}  
}

void display(void)
{

do 2-D OpenGL graphics.

z point is ROTATED spin degrees AROUNd the z

transformation so every subsequent world coordinate

getRotatef(spin, 0.0, 0.0, 1.0);  

getRotatef();  

Sometimes getUniformLocation will register a callback

static GLfloat spin = 0.0;

4

University at Albany Computer Science Dept.
function END OF OUR display() { (don't wait).

frame; get it started on the new frame; and return

Wait for the system to finish drawing the previous

flushbuffers();

BEFORE the rotation was installed.

RESTORE the current transformation to what it was

popmatrix();

square to render.

TRANSFORMS the model square into the rotated

The transformation in effect, rotate-by-spin,
function as modeling coordinates; that's our model.
DRAW A SQUARE: (-25.0, -25.0) and (25.0, 25.0)

getrect(-25.0, -25.0, 25.0, 25.0);

getcolor(1.0, 1.0, 1.0);
GLoadIdentity();
GLMatrixMode(GL_PROJECTION);
GLLoadIdentity (0, 0, (GLint) m, (GLint) N);

void reshape(int w, int h) {

void int (void)

{

void int (void)

{

GLMatrixMode (GL_FLAT);

GLClearColor (0.0, 0.0, 0.0, 0.0);

)

}

}

GLutPostRedisplay();

if (unps < 360.0)
unps = unps + 2.0;

)

}

}

void spindisplay(void)

University at Albany Computer Science Dept.
break;  
  
default:  
break;

switch (button)  
}  

void mouse(int button, int state, int x, int y)  
{

  
}
```c
int main(int argc, char** argv) {

    glutInitLoop();
    glutMouseFunc(mouse);
    glut reshapeFunc(reshape);
    glutDisplayFunc(display);

    init();

    glutCreateWindow (argc, argv[0]);
    glutInitWindowPosition (100, 100);
    glutInitWindowSize (250, 250);
    glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
    glutInit (argc, argv);

    }/*
*
Register mouse input callback functions
* Request double buffer display mode.
*/
```
return 0;
*/ANSI C requires main to return int.*
}
and resolutions.

Independent of the device details, such as monitor and window sizes.

- Calculations of normalized coordinates from world coordinates are independent of the # of pixels.
- Defined so the range is [0,1] or [-1,1].
- Coordinates from world coordinates.

normalized coordinates: Use in the calculations of screen coordinates.

Always unsigned integers!

Range: 0 to (#horiz. pixels-1, #vert. pixels-1).

device or screen coordinates: 1 unit corresponds to one pixel.
begin a connected series of one OR MORE lines.

OpenGL also uses one endpoint per line, but with absolute coordinates.

- have to be passed twice.

only one endpoint per function call is needed; the same coordinates don’t

Hence, to draw a connected series of lines using relative coordinates,

draw a line changes the current position to the absolute position of

the new endpoint.

Drawing a line changes the current position to the absolute position of

the world coordinate system) and sometimes current direction.

primitive drawing functions use offsets from the current position.

In the turtle graphics API style (of LOGO, etc., and Postscript), the

- functions expect absolute coordinates.

In the world, in normalized space, or in the screen, OpenGL primitive

- Absolute coordinates are meaningful in terms of a fixed coordinate frame.

- Absolute versus relative coordinates:
world will be viewed, and how the camera works.

```csharp
GLMatrixMode(GL_PROJECTION);

GLLoadIdentity();

GLOrtho2D(0.0, 200.0, 0.0, 150.0);  
```
Include this addition in the Bresenham loop:

(a) Calculate the memory address strides—number of bytes between adjacent entries in an array (both horizontal and vertical).
(b) Add the appropriate stride to calculate the address of the next access.
(c) Include this addition in the Bresenham loop!

Assembly Language Ideas:

3. Setting Frame Buffer Values: (H# 3-7) Implementing array access in Modern OpenGL supports graphics cards with multiple processors on processors, rasterize the parts simultaneously.

2. Parallel Processing: (H# 3-6) Divide up the work so multiple hardware processors parallel processing polylines: Don’t draw endpoints twice. (end of H# 3-7).

Miscellaneous, easy details:
5. (HB 3-9, 10, 11) Rasterization of various kinds of curves.

enough
also can draw curved things with polylines whose segments are small
OpenGL and GLU to draw splines or polylines. Graphics programmers
NOT circles as primitives. GLUT draws curved things by calling
OpenGL draws lines and certain polynomial spline curves,
Angle $\Theta$ ranges over the whole circle.

3. Parametric: $x = (\theta)\cos(\theta), y = (\theta)\sin(\theta)$.

2. Implicit: $x^2 + y^2 = \theta^2$.

1. Function to plot: $x^2 + y^2 = \theta^2$ and $xy = \theta$.

3. Mathematical expressions for curves:
Three separate numeric scalar addition define what vector

\[(zq + z'q, q' + h'q, xq + x'q) = (zq, q', xq) + (z', h', x')\]

Vector addition:

- Often drawn as an arrow.
- Represent a displacement (or velocity).
- Locate a point.

More subtle—Two ways to use vectors.

When a coordinate system (origin, axes directions, and unit length) is defined in a plane or 3d space, vectors locate points therein.

When a coordinate system (origin, axes directions, and unit length) is defined in a plane or 3d space, vectors locate points therein.

Components.

For graphics, for now, our vectors will have two or three numbers. For generic computer programmers, a vector is, very concretely, an array of

But first, we review/introduce vectors.
Let's examine the house example from the HOMEWORK:

1. a > 0, shrinks length if |a| < 1.
2. Stretches a displacement's length by a factor of a (reverses direction if transformation).

I. "Blows up" (enlarges) the world by a factor of a (uniform scaling).

Basic geometric significance:

\((a, b, c, d) \rightarrow (ax, ay, az) = \alpha \cdot (x, y, z)\)

Vector-scalar multiplication: \(\alpha\) is a number.

2. Tail-to-head composition of displacements.

1. Apply a displacement to a point to locate another point.

Basic geometric significance: