

# GL: 3-D Graphics

Version 7.7.0.901

Scott Owens

July 30, 2020

The `sgl` libraries provide access to the rendering functions of OpenGL 1.5 and GLU 1.3 libraries. The `sgl` libraries do not address system-level concerns, such as the attachment of GL rendering contexts to displays. Instead, the libraries should work with any Racket extension that provides GL with access to the system (such as a binding for `glx`). Notably, the `racket/gui/base` library provides support for rendering contexts via the `canvas%` class and its `with-gl-context` method.

## **Contents**

<b>1</b>	<b>Using OpenGL</b>	<b>3</b>
<b>2</b>	<b>C-Style OpenGL</b>	<b>4</b>
<b>3</b>	<b>Racket-Style OpenGL</b>	<b>40</b>
<b>4</b>	<b>OpenGL Vectors</b>	<b>45</b>
<b>5</b>	<b>Bitmaps</b>	<b>51</b>
<b>6</b>	<b>Initialization</b>	<b>52</b>
	<b>Index</b>	<b>53</b>
	<b>Index</b>	<b>53</b>

# 1 Using OpenGL

The `sgl/gl` library provides direct access to the C-style OpenGL API, whereas the `sgl` library provides a more Racket-like interface. The `sgl/gl` library provides a binding for each `#defined` constant and for most functions in OpenGL 1.5 and GLU 1.3. The functions perform comparable checking to their C-language counterparts; they check the types of their arguments, but do not check the length of array arguments. The `sgl` library provides wrappers around many of the functions in the `sgl/gl` library to present a more Racket-friendly interface, including function names that follow Racket conventions, and checked, symbolic enumeration arguments, and array-length checks.

**Warning on Safety:** OpenGL programming is inherently unsafe, even when using only the `sgl` library. Although `sgl` checks the arguments to each function call, violation of higher-level assumptions of the system's OpenGL library can cause it to crash, bringing the entire Racket system down. For example, sending a large number of vertices in a single `glBegin` causes at least some GL implementations to crash.

Some examples are available in the "examples" directory of the "sgl" collection in the Racket installation. For "alpha.rkt", try choosing the "sk.jpg" image distributed with Racket in the "icons" collection; you may have to press the `t` key a few times if the spinning cube is blank.

## 2 C-Style OpenGL

(require sgl/gl) package: sgl

The `sgl/gl` module provides a direct interface to the system's GL library closely following the conventions of the C-language OpenGL API. It provides a binding for each `#defined` constant (these start with `GL_`) and for the functions in the GL 1.5 and GLU 1.3 specifications, except for the following:

- Vertex arrays (GL 1.5, Section 2.8)
- Buffer objects (GL 1.5, Section 2.9)
- `glGetPointerv` (GL 1.5, Section 6.1.11)
- Buffer object queries (GL 1.5, Section 6.1.13)
- Polygon tessellation (GLU 1.3, Section 5)
- `gluQuadricCallback` (GLU 1.3, Section 6.2)
- NURBS callbacks (GLU 1.3, Section 7.2)

If one of the provided functions is not present on your system (e.g. if your system supports only GL 1.3), then the corresponding `sgl/gl` function raises a run-time exception when invoked.

The functions provided by `sgl/gl` perform comparable checking to their C-language counterparts; they check the types of their arguments, but do not check the length of array arguments. The following details the kinds of Racket values that can be provided for each primitive OpenGL type:

- `GLbyte`, `GLshort`, `GLint`: exact integer in the proper range
- `GLubyte`, `GLushort`, `GLuint`: exact non-negative integer in the proper range
- `GLsizei`, `GLenum`, `GLbitfield`: exact non-negative integer in the proper range
- `GLfloat`, `GLdouble`: real number
- `GLclampf`, `GLclampd`: real number
- `GLboolean`: any value, where `#f` means `GL_FALSE` and all other values mean `GL_TRUE`; do not use `GL_FALSE` or `GL_TRUE`, since they are bound to integers, both will end up being converted to `GL_TRUE`.

OpenGL functions that take vector arguments accept `cvector` values. The type of the `cvector` is checked; for example, `glVertex3fv` expects a vector of `GLfloat`s, so `glVertex3fv` accepts only a `cvector` containing reals. See also `sgl/gl-vectors`. Functions that accept arrays of type `void*` accept any `cvector`; you must ensure that you supply the proper kind of vector, as in the C-language OpenGL API.

Examples:

```
(require sgl/gl
        sgl/gl-vectors)
(glBegin GL_TRIANGLES)
(glVertex3i 1 2 3)
(glVertex4fv (gl-float-vector 1 2 3 4))
(glEnd)
```

```
glPixelMapfv : procedure?
glPixelMapuiv : procedure?
glPixelMapusv : procedure?
glDeleteTextures : procedure?
glDeleteQueries : procedure?
```

These functions do not take a size argument, because it is derived from the length of the argument vector.

```
glGenTextures : procedure?
glGenQueries : procedure?
```

These functions do not take vector arguments. Instead, they allocate a vector of the requested size and return it.

```
glAreTexturesResident : procedure?
```

This function takes in a `GLuint` vector and textures, and it returns 2 values: the specified boolean and a boolean vector of residences.

```
glGetBooleanv : procedure?
glGetIntegerv : procedure?
glGetFloatv : procedure?
glGetDoublev : procedure?
glGetLightfv : procedure?
glGetLightiv : procedure?
glGetMaterialfv : procedure?
glGetMaterialiv : procedure?
glGetTexEnvfv : procedure?
glGetTexEnviv : procedure?
```

```
glGetTexGendv : procedure?  
glGetTexGenfv : procedure?  
glGetTexGeniv : procedure?  
glGetTexParameterfv : procedure?  
glGetTexParameteriv : procedure?  
glGetTexLevelParameterfv : procedure?  
glGetTexLevelParameteriv : procedure?  
glGetPixelMapfv : procedure?  
glGetPixelMapuiv : procedure?  
glGetPixelMapusv : procedure?  
glGetMapdv : procedure?  
glGetMapfv : procedure?  
glGetMapiv : procedure?  
glGetBufferParameteriv : procedure?  
glGetConvolutionParameterfv : procedure?  
glGetConvolutionParameteriv : procedure?  
glGetHistogramParameterfv : procedure?  
glGetHistogramParameteriv : procedure?  
glGetMinmaxParameterfv : procedure?  
glGetMinmaxParameteriv : procedure?  
glGetQueryiv : procedure?  
glGetQueryObjectiv : procedure?  
glGetQueryObjectuiv : procedure?
```

Instead of taking a vector argument, these function take an integer argument that specifies the size of the vector that is returned.

```
glGetClipPlane : procedure?
```

This function does not take a vector argument and returns a GLdouble vector of length 4.

```
glGetString : procedure?  
gluCheckExtension : procedure?  
gluErrorString : procedure?  
gluGetString : procedure?
```

These functions deal with strings instead of GLubyte vectors.

```
gluProject : procedure?  
gluUnProject : procedure?  
gluUnProject4 : procedure?
```

Instead of taking pointers to GLdoubles for return values, these function directly return GLdouble vectors.

```
glSelectBuffer : procedure?  
glFeedbackBuffer : procedure?
```

These functions do not take vectors, instead they return a `selection-buffer-object` or `feedback-buffer-object`. The `select-buffer->gl-uint-vector` and `feedback-buffer->gl-float-vector` functions copy the contents of the buffer into a vector. Because the OpenGL library writes to the buffer-object on OpenGL function calls after `glSelectBuffer` or `glFeedbackBuffer` has returned, if the buffer is garbage collected before OpenGL is finished writing to it, the entire Racket system can crash. The `gl-process-selection` function in `sgl` helps interpret the results of `glSelectBuffer` in a Racket-friendly format.

```
glAccum : procedure?  
glActiveTexture : procedure?  
glAlphaFunc : procedure?  
glBegin : procedure?  
glBeginQuery : procedure?  
glBindTexture : procedure?  
glBitmap : procedure?  
glBlendColor : procedure?  
glBlendEquation : procedure?  
glBlendFunc : procedure?  
glBlendFuncSeparate : procedure?  
glCallList : procedure?  
glCallLists : procedure?  
glClear : procedure?  
glClearAccum : procedure?  
glClearColor : procedure?  
glClearDepth : procedure?  
glClearIndex : procedure?  
glClearStencil : procedure?  
glClipPlane : procedure?  
glColor3b : procedure?  
glColor3bv : procedure?  
glColor3d : procedure?  
glColor3dv : procedure?  
glColor3f : procedure?  
glColor3fv : procedure?  
glColor3i : procedure?  
glColor3iv : procedure?  
glColor3s : procedure?  
glColor3sv : procedure?  
glColor3ub : procedure?  
glColor3ubv : procedure?
```

glColor3ui : procedure?  
glColor3uiv : procedure?  
glColor3us : procedure?  
glColor3usv : procedure?  
glColor4b : procedure?  
glColor4bv : procedure?  
glColor4d : procedure?  
glColor4dv : procedure?  
glColor4f : procedure?  
glColor4fv : procedure?  
glColor4i : procedure?  
glColor4iv : procedure?  
glColor4s : procedure?  
glColor4sv : procedure?  
glColor4ub : procedure?  
glColor4ubv : procedure?  
glColor4ui : procedure?  
glColor4uiv : procedure?  
glColor4us : procedure?  
glColor4usv : procedure?  
glColorMask : procedure?  
glColorMaterial : procedure?  
glColorSubTable : procedure?  
glColorTable : procedure?  
glColorTableParameterfv : procedure?  
glColorTableParameteriv : procedure?  
glCompressedTexImage1D : procedure?  
glCompressedTexImage2D : procedure?  
glCompressedTexImage3D : procedure?  
glCompressedTexSubImage1D : procedure?  
glCompressedTexSubImage2D : procedure?  
glCompressedTexSubImage3D : procedure?  
glConvolutionFilter1D : procedure?  
glConvolutionFilter2D : procedure?  
glConvolutionParameterf : procedure?  
glConvolutionParameterfv : procedure?  
glConvolutionParameteri : procedure?  
glConvolutionParameteriv : procedure?  
glCopyColorSubTable : procedure?  
glCopyColorTable : procedure?  
glCopyConvolutionFilter1D : procedure?  
glCopyConvolutionFilter2D : procedure?  
glCopyPixels : procedure?  
glCopyTexImage1D : procedure?



glCopyTexImage2D : procedure?  
glCopyTexSubImage1D : procedure?  
glCopyTexSubImage2D : procedure?  
glCopyTexSubImage3D : procedure?  
glCullFace : procedure?  
glDeleteLists : procedure?  
glDepthFunc : procedure?  
glDepthMask : procedure?  
glDepthRange : procedure?  
glDisable : procedure?  
glDrawBuffer : procedure?  
glDrawPixels : procedure?  
glEdgeFlag : procedure?  
glEdgeFlagv : procedure?  
glEnable : procedure?  
glEnd : procedure?  
glEndList : procedure?  
glEndQuery : procedure?  
glEvalCoord1d : procedure?  
glEvalCoord1dv : procedure?  
glEvalCoord1f : procedure?  
glEvalCoord1fv : procedure?  
glEvalCoord2d : procedure?  
glEvalCoord2dv : procedure?  
glEvalCoord2f : procedure?  
glEvalCoord2fv : procedure?  
glEvalMesh1 : procedure?  
glEvalMesh2 : procedure?  
glEvalPoint1 : procedure?  
glEvalPoint2 : procedure?  
glFinish : procedure?  
glFlush : procedure?  
glFogCoordd : procedure?  
glFogCoorddv : procedure?  
glFogCoordf : procedure?  
glFogCoordfv : procedure?  
glFogf : procedure?  
glFogfv : procedure?  
glFogi : procedure?  
glFogiv : procedure?  
glFrontFace : procedure?  
glFrustum : procedure?  
glGenLists : procedure?  
glGetColorTable : procedure?

glGetCompressedTexImage : procedure?  
glGetConvolutionFilter : procedure?  
glGetError : procedure?  
glGetHistogram : procedure?  
glGetMinmax : procedure?  
glGetPolygonStipple : procedure?  
glGetSeparableFilter : procedure?  
glGetTexImage : procedure?  
glHint : procedure?  
glHistogram : procedure?  
glIndexMask : procedure?  
glIndexd : procedure?  
glIndexdv : procedure?  
glIndexf : procedure?  
glIndexfv : procedure?  
glIndexi : procedure?  
glIndexiv : procedure?  
glIndexs : procedure?  
glIndexsv : procedure?  
glIndexub : procedure?  
glIndexubv : procedure?  
glInitNames : procedure?  
glIsBuffer : procedure?  
glIsEnabled : procedure?  
glIsList : procedure?  
glIsQuery : procedure?  
glIsTexture : procedure?  
glLightModelf : procedure?  
glLightModelfv : procedure?  
glLightModeli : procedure?  
glLightModeliv : procedure?  
glLightf : procedure?  
glLightfv : procedure?  
glLighti : procedure?  
glLightiv : procedure?  
glLineStipple : procedure?  
glLineWidth : procedure?  
glListBase : procedure?  
glLoadIdentity : procedure?  
glLoadMatrixd : procedure?  
glLoadMatrixf : procedure?  
glLoadName : procedure?  
glLoadTransposeMatrixd : procedure?  
glLoadTransposeMatrixf : procedure?

glLogicOp : procedure?  
glMap1d : procedure?  
glMap1f : procedure?  
glMap2d : procedure?  
glMap2f : procedure?  
glMapGrid1d : procedure?  
glMapGrid1f : procedure?  
glMapGrid2d : procedure?  
glMapGrid2f : procedure?  
glMaterialf : procedure?  
glMaterialfv : procedure?  
glMateriali : procedure?  
glMaterialiv : procedure?  
glMatrixMode : procedure?  
glMinmax : procedure?  
glMultMatrixd : procedure?  
glMultMatrixf : procedure?  
glMultTransposeMatrixd : procedure?  
glMultTransposeMatrixf : procedure?  
glMultiTexCoord1d : procedure?  
glMultiTexCoord1dv : procedure?  
glMultiTexCoord1f : procedure?  
glMultiTexCoord1fv : procedure?  
glMultiTexCoord1i : procedure?  
glMultiTexCoord1iv : procedure?  
glMultiTexCoord1s : procedure?  
glMultiTexCoord1sv : procedure?  
glMultiTexCoord2d : procedure?  
glMultiTexCoord2dv : procedure?  
glMultiTexCoord2f : procedure?  
glMultiTexCoord2fv : procedure?  
glMultiTexCoord2i : procedure?  
glMultiTexCoord2iv : procedure?  
glMultiTexCoord2s : procedure?  
glMultiTexCoord2sv : procedure?  
glMultiTexCoord3d : procedure?  
glMultiTexCoord3dv : procedure?  
glMultiTexCoord3f : procedure?  
glMultiTexCoord3fv : procedure?  
glMultiTexCoord3i : procedure?  
glMultiTexCoord3iv : procedure?  
glMultiTexCoord3s : procedure?  
glMultiTexCoord3sv : procedure?  
glMultiTexCoord4d : procedure?

glMultiTexCoord4dv : procedure?  
glMultiTexCoord4f : procedure?  
glMultiTexCoord4fv : procedure?  
glMultiTexCoord4i : procedure?  
glMultiTexCoord4iv : procedure?  
glMultiTexCoord4s : procedure?  
glMultiTexCoord4sv : procedure?  
glNewList : procedure?  
glNormal3b : procedure?  
glNormal3bv : procedure?  
glNormal3d : procedure?  
glNormal3dv : procedure?  
glNormal3f : procedure?  
glNormal3fv : procedure?  
glNormal3i : procedure?  
glNormal3iv : procedure?  
glNormal3s : procedure?  
glNormal3sv : procedure?  
glOrtho : procedure?  
glPassThrough : procedure?  
glPixelStoref : procedure?  
glPixelStorei : procedure?  
glPixelTransferf : procedure?  
glPixelTransferi : procedure?  
glPixelZoom : procedure?  
glPointParameterf : procedure?  
glPointParameterfv : procedure?  
glPointParameteri : procedure?  
glPointParameteriv : procedure?  
glPointSize : procedure?  
glPolygonMode : procedure?  
glPolygonOffset : procedure?  
glPolygonStipple : procedure?  
glPopAttrib : procedure?  
glPopClientAttrib : procedure?  
glPopMatrix : procedure?  
glPopName : procedure?  
glPushAttrib : procedure?  
glPushClientAttrib : procedure?  
glPushMatrix : procedure?  
glPushName : procedure?  
glRasterPos2d : procedure?  
glRasterPos2dv : procedure?  
glRasterPos2f : procedure?

glRasterPos2fv : procedure?  
glRasterPos2i : procedure?  
glRasterPos2iv : procedure?  
glRasterPos2s : procedure?  
glRasterPos2sv : procedure?  
glRasterPos3d : procedure?  
glRasterPos3dv : procedure?  
glRasterPos3f : procedure?  
glRasterPos3fv : procedure?  
glRasterPos3i : procedure?  
glRasterPos3iv : procedure?  
glRasterPos3s : procedure?  
glRasterPos3sv : procedure?  
glRasterPos4d : procedure?  
glRasterPos4dv : procedure?  
glRasterPos4f : procedure?  
glRasterPos4fv : procedure?  
glRasterPos4i : procedure?  
glRasterPos4iv : procedure?  
glRasterPos4s : procedure?  
glRasterPos4sv : procedure?  
glReadBuffer : procedure?  
glReadPixels : procedure?  
glRectd : procedure?  
glRectdv : procedure?  
glRectf : procedure?  
glRectfv : procedure?  
glRecti : procedure?  
glRectiv : procedure?  
glRects : procedure?  
glRectsv : procedure?  
glRenderMode : procedure?  
glResetHistogram : procedure?  
glResetMinmax : procedure?  
glRotated : procedure?  
glRotatef : procedure?  
glSampleCoverage : procedure?  
glScaled : procedure?  
glScalef : procedure?  
glScissor : procedure?  
glSecondaryColor3b : procedure?  
glSecondaryColor3bv : procedure?  
glSecondaryColor3d : procedure?  
glSecondaryColor3dv : procedure?

glSecondaryColor3f : procedure?  
glSecondaryColor3fv : procedure?  
glSecondaryColor3i : procedure?  
glSecondaryColor3iv : procedure?  
glSecondaryColor3s : procedure?  
glSecondaryColor3sv : procedure?  
glSecondaryColor3ub : procedure?  
glSecondaryColor3ubv : procedure?  
glSecondaryColor3ui : procedure?  
glSecondaryColor3uiv : procedure?  
glSecondaryColor3us : procedure?  
glSecondaryColor3usv : procedure?  
glSeparableFilter2D : procedure?  
glShadeModel : procedure?  
glStencilFunc : procedure?  
glStencilMask : procedure?  
glStencilOp : procedure?  
glTexCoord1d : procedure?  
glTexCoord1dv : procedure?  
glTexCoord1f : procedure?  
glTexCoord1fv : procedure?  
glTexCoord1i : procedure?  
glTexCoord1iv : procedure?  
glTexCoord1s : procedure?  
glTexCoord1sv : procedure?  
glTexCoord2d : procedure?  
glTexCoord2dv : procedure?  
glTexCoord2f : procedure?  
glTexCoord2fv : procedure?  
glTexCoord2i : procedure?  
glTexCoord2iv : procedure?  
glTexCoord2s : procedure?  
glTexCoord2sv : procedure?  
glTexCoord3d : procedure?  
glTexCoord3dv : procedure?  
glTexCoord3f : procedure?  
glTexCoord3fv : procedure?  
glTexCoord3i : procedure?  
glTexCoord3iv : procedure?  
glTexCoord3s : procedure?  
glTexCoord3sv : procedure?  
glTexCoord4d : procedure?  
glTexCoord4dv : procedure?  
glTexCoord4f : procedure?

glTexCoord4fv : procedure?  
glTexCoord4i : procedure?  
glTexCoord4iv : procedure?  
glTexCoord4s : procedure?  
glTexCoord4sv : procedure?  
glTexEnvf : procedure?  
glTexEnvfv : procedure?  
glTexEnvi : procedure?  
glTexEnviv : procedure?  
glTexGend : procedure?  
glTexGendv : procedure?  
glTexGenf : procedure?  
glTexGenfv : procedure?  
glTexGeni : procedure?  
glTexGeniv : procedure?  
glTexImage1D : procedure?  
glTexImage2D : procedure?  
glTexImage3D : procedure?  
glTexParameterf : procedure?  
glTexParameterfv : procedure?  
glTexParameteri : procedure?  
glTexParameteriv : procedure?  
glTexSubImage1D : procedure?  
glTexSubImage2D : procedure?  
glTexSubImage3D : procedure?  
glTranslated : procedure?  
glTranslatef : procedure?  
glVertex2d : procedure?  
glVertex2dv : procedure?  
glVertex2f : procedure?  
glVertex2fv : procedure?  
glVertex2i : procedure?  
glVertex2iv : procedure?  
glVertex2s : procedure?  
glVertex2sv : procedure?  
glVertex3d : procedure?  
glVertex3dv : procedure?  
glVertex3f : procedure?  
glVertex3fv : procedure?  
glVertex3i : procedure?  
glVertex3iv : procedure?  
glVertex3s : procedure?  
glVertex3sv : procedure?  
glVertex4d : procedure?

glVertex4dv : procedure?  
glVertex4f : procedure?  
glVertex4fv : procedure?  
glVertex4i : procedure?  
glVertex4iv : procedure?  
glVertex4s : procedure?  
glVertex4sv : procedure?  
glViewport : procedure?  
glWindowPos2d : procedure?  
glWindowPos2dv : procedure?  
glWindowPos2f : procedure?  
glWindowPos2fv : procedure?  
glWindowPos2i : procedure?  
glWindowPos2iv : procedure?  
glWindowPos2s : procedure?  
glWindowPos2sv : procedure?  
glWindowPos3d : procedure?  
glWindowPos3dv : procedure?  
glWindowPos3f : procedure?  
glWindowPos3fv : procedure?  
glWindowPos3i : procedure?  
glWindowPos3iv : procedure?  
glWindowPos3s : procedure?  
glWindowPos3sv : procedure?  
gluBuild1DMipmapLevels : procedure?  
gluBuild1DMipmaps : procedure?  
gluBuild2DMipmapLevels : procedure?  
gluBuild2DMipmaps : procedure?  
gluBuild3DMipmapLevels : procedure?  
gluBuild3DMipmaps : procedure?  
gluCylinder : procedure?  
gluDisk : procedure?  
gluLookAt : procedure?  
gluNewQuadric : procedure?  
gluOrtho2D : procedure?  
gluPartialDisk : procedure?  
gluPerspective : procedure?  
gluPickMatrix : procedure?  
gluQuadricDrawStyle : procedure?  
gluQuadricNormals : procedure?  
gluQuadricOrientation : procedure?  
gluQuadricTexture : procedure?  
gluScaleImage : procedure?  
gluSphere : procedure?



These functions are all direct translations of the C OpenGL API.

```
GL_FALSE : exact-integer?  
GL_TRUE : exact-integer?  
GL_BYTE : exact-integer?  
GL_UNSIGNED_BYTE : exact-integer?  
GL_SHORT : exact-integer?  
GL_UNSIGNED_SHORT : exact-integer?  
GL_INT : exact-integer?  
GL_UNSIGNED_INT : exact-integer?  
GL_FLOAT : exact-integer?  
GL_DOUBLE : exact-integer?  
GL_2_BYTES : exact-integer?  
GL_3_BYTES : exact-integer?  
GL_4_BYTES : exact-integer?  
GL_POINTS : exact-integer?  
GL_LINES : exact-integer?  
GL_LINE_LOOP : exact-integer?  
GL_LINE_STRIP : exact-integer?  
GL_TRIANGLES : exact-integer?  
GL_TRIANGLE_STRIP : exact-integer?  
GL_TRIANGLE_FAN : exact-integer?  
GL_QUADS : exact-integer?  
GL_QUAD_STRIP : exact-integer?  
GL_POLYGON : exact-integer?  
GL_VERTEX_ARRAY : exact-integer?  
GL_NORMAL_ARRAY : exact-integer?  
GL_COLOR_ARRAY : exact-integer?  
GL_INDEX_ARRAY : exact-integer?  
GL_TEXTURE_COORD_ARRAY : exact-integer?  
GL_EDGE_FLAG_ARRAY : exact-integer?  
GL_VERTEX_ARRAY_SIZE : exact-integer?  
GL_VERTEX_ARRAY_TYPE : exact-integer?  
GL_VERTEX_ARRAY_STRIDE : exact-integer?  
GL_NORMAL_ARRAY_TYPE : exact-integer?  
GL_NORMAL_ARRAY_STRIDE : exact-integer?  
GL_COLOR_ARRAY_SIZE : exact-integer?  
GL_COLOR_ARRAY_TYPE : exact-integer?  
GL_COLOR_ARRAY_STRIDE : exact-integer?  
GL_INDEX_ARRAY_TYPE : exact-integer?  
GL_INDEX_ARRAY_STRIDE : exact-integer?  
GL_TEXTURE_COORD_ARRAY_SIZE : exact-integer?  
GL_TEXTURE_COORD_ARRAY_TYPE : exact-integer?  
GL_TEXTURE_COORD_ARRAY_STRIDE : exact-integer?  
GL_EDGE_FLAG_ARRAY_STRIDE : exact-integer?
```

GL\_VERTEX\_ARRAY\_POINTER : exact-integer?  
GL\_NORMAL\_ARRAY\_POINTER : exact-integer?  
GL\_COLOR\_ARRAY\_POINTER : exact-integer?  
GL\_INDEX\_ARRAY\_POINTER : exact-integer?  
GL\_TEXTURE\_COORD\_ARRAY\_POINTER : exact-integer?  
GL\_EDGE\_FLAG\_ARRAY\_POINTER : exact-integer?  
GL\_V2F : exact-integer?  
GL\_V3F : exact-integer?  
GL\_C4UB\_V2F : exact-integer?  
GL\_C4UB\_V3F : exact-integer?  
GL\_C3F\_V3F : exact-integer?  
GL\_N3F\_V3F : exact-integer?  
GL\_C4F\_N3F\_V3F : exact-integer?  
GL\_T2F\_V3F : exact-integer?  
GL\_T4F\_V4F : exact-integer?  
GL\_T2F\_C4UB\_V3F : exact-integer?  
GL\_T2F\_C3F\_V3F : exact-integer?  
GL\_T2F\_N3F\_V3F : exact-integer?  
GL\_T2F\_C4F\_N3F\_V3F : exact-integer?  
GL\_T4F\_C4F\_N3F\_V4F : exact-integer?  
GL\_MATRIX\_MODE : exact-integer?  
GL\_MODELVIEW : exact-integer?  
GL\_PROJECTION : exact-integer?  
GL\_TEXTURE : exact-integer?  
GL\_POINT\_SMOOTH : exact-integer?  
GL\_POINT\_SIZE : exact-integer?  
GL\_POINT\_SIZE\_GRANULARITY : exact-integer?  
GL\_POINT\_SIZE\_RANGE : exact-integer?  
GL\_LINE\_SMOOTH : exact-integer?  
GL\_LINE\_STIPPLE : exact-integer?  
GL\_LINE\_STIPPLE\_PATTERN : exact-integer?  
GL\_LINE\_STIPPLE\_REPEAT : exact-integer?  
GL\_LINE\_WIDTH : exact-integer?  
GL\_LINE\_WIDTH\_GRANULARITY : exact-integer?  
GL\_LINE\_WIDTH\_RANGE : exact-integer?  
GL\_POINT : exact-integer?  
GL\_LINE : exact-integer?  
GL\_FILL : exact-integer?  
GL\_CW : exact-integer?  
GL\_CCW : exact-integer?  
GL\_FRONT : exact-integer?  
GL\_BACK : exact-integer?  
GL\_POLYGON\_MODE : exact-integer?  
GL\_POLYGON\_SMOOTH : exact-integer?

GL\_POLYGON\_STIPPLE : exact-integer?  
GL\_EDGE\_FLAG : exact-integer?  
GL\_CULL\_FACE : exact-integer?  
GL\_CULL\_FACE\_MODE : exact-integer?  
GL\_FRONT\_FACE : exact-integer?  
GL\_POLYGON\_OFFSET\_FACTOR : exact-integer?  
GL\_POLYGON\_OFFSET\_UNITS : exact-integer?  
GL\_POLYGON\_OFFSET\_POINT : exact-integer?  
GL\_POLYGON\_OFFSET\_LINE : exact-integer?  
GL\_POLYGON\_OFFSET\_FILL : exact-integer?  
GL\_COMPILE : exact-integer?  
GL\_COMPILE\_AND\_EXECUTE : exact-integer?  
GL\_LIST\_BASE : exact-integer?  
GL\_LIST\_INDEX : exact-integer?  
GL\_LIST\_MODE : exact-integer?  
GL\_NEVER : exact-integer?  
GL\_LESS : exact-integer?  
GL\_EQUAL : exact-integer?  
GL\_LEQUAL : exact-integer?  
GL\_GREATER : exact-integer?  
GL\_NOTEQUAL : exact-integer?  
GL\_GEQUAL : exact-integer?  
GL\_ALWAYS : exact-integer?  
GL\_DEPTH\_TEST : exact-integer?  
GL\_DEPTH\_BITS : exact-integer?  
GL\_DEPTH\_CLEAR\_VALUE : exact-integer?  
GL\_DEPTH\_FUNC : exact-integer?  
GL\_DEPTH\_RANGE : exact-integer?  
GL\_DEPTH\_WRITEMASK : exact-integer?  
GL\_DEPTH\_COMPONENT : exact-integer?  
GL\_LIGHTING : exact-integer?  
GL\_LIGHT0 : exact-integer?  
GL\_LIGHT1 : exact-integer?  
GL\_LIGHT2 : exact-integer?  
GL\_LIGHT3 : exact-integer?  
GL\_LIGHT4 : exact-integer?  
GL\_LIGHT5 : exact-integer?  
GL\_LIGHT6 : exact-integer?  
GL\_LIGHT7 : exact-integer?  
GL\_SPOT\_EXPONENT : exact-integer?  
GL\_SPOT\_CUTOFF : exact-integer?  
GL\_CONSTANT\_ATTENUATION : exact-integer?  
GL\_LINEAR\_ATTENUATION : exact-integer?  
GL\_QUADRATIC\_ATTENUATION : exact-integer?

GL\_AMBIENT : exact-integer?  
GL\_DIFFUSE : exact-integer?  
GL\_SPECULAR : exact-integer?  
GL\_SHININESS : exact-integer?  
GL\_EMISSION : exact-integer?  
GL\_POSITION : exact-integer?  
GL\_SPOT\_DIRECTION : exact-integer?  
GL\_AMBIENT\_AND\_DIFFUSE : exact-integer?  
GL\_COLOR\_INDEXES : exact-integer?  
GL\_LIGHT\_MODEL\_TWO\_SIDE : exact-integer?  
GL\_LIGHT\_MODEL\_LOCAL\_VIEWER : exact-integer?  
GL\_LIGHT\_MODEL\_AMBIENT : exact-integer?  
GL\_FRONT\_AND\_BACK : exact-integer?  
GL\_SHADE\_MODEL : exact-integer?  
GL\_FLAT : exact-integer?  
GL\_SMOOTH : exact-integer?  
GL\_COLOR\_MATERIAL : exact-integer?  
GL\_COLOR\_MATERIAL\_FACE : exact-integer?  
GL\_COLOR\_MATERIAL\_PARAMETER : exact-integer?  
GL\_NORMALIZE : exact-integer?  
GL\_CLIP\_PLANE0 : exact-integer?  
GL\_CLIP\_PLANE1 : exact-integer?  
GL\_CLIP\_PLANE2 : exact-integer?  
GL\_CLIP\_PLANE3 : exact-integer?  
GL\_CLIP\_PLANE4 : exact-integer?  
GL\_CLIP\_PLANE5 : exact-integer?  
GL\_ACCUM\_RED\_BITS : exact-integer?  
GL\_ACCUM\_GREEN\_BITS : exact-integer?  
GL\_ACCUM\_BLUE\_BITS : exact-integer?  
GL\_ACCUM\_ALPHA\_BITS : exact-integer?  
GL\_ACCUM\_CLEAR\_VALUE : exact-integer?  
GL\_ACCUM : exact-integer?  
GL\_ADD : exact-integer?  
GL\_LOAD : exact-integer?  
GL\_MULT : exact-integer?  
GL\_RETURN : exact-integer?  
GL\_ALPHA\_TEST : exact-integer?  
GL\_ALPHA\_TEST\_REF : exact-integer?  
GL\_ALPHA\_TEST\_FUNC : exact-integer?  
GL\_BLEND : exact-integer?  
GL\_BLEND\_SRC : exact-integer?  
GL\_BLEND\_DST : exact-integer?  
GL\_ZERO : exact-integer?  
GL\_ONE : exact-integer?

GL\_SRC\_COLOR : exact-integer?  
GL\_ONE\_MINUS\_SRC\_COLOR : exact-integer?  
GL\_SRC\_ALPHA : exact-integer?  
GL\_ONE\_MINUS\_SRC\_ALPHA : exact-integer?  
GL\_DST\_ALPHA : exact-integer?  
GL\_ONE\_MINUS\_DST\_ALPHA : exact-integer?  
GL\_DST\_COLOR : exact-integer?  
GL\_ONE\_MINUS\_DST\_COLOR : exact-integer?  
GL\_SRC\_ALPHA\_SATURATE : exact-integer?  
GL\_FEEDBACK : exact-integer?  
GL\_RENDER : exact-integer?  
GL\_SELECT : exact-integer?  
GL\_2D : exact-integer?  
GL\_3D : exact-integer?  
GL\_3D\_COLOR : exact-integer?  
GL\_3D\_COLOR\_TEXTURE : exact-integer?  
GL\_4D\_COLOR\_TEXTURE : exact-integer?  
GL\_POINT\_TOKEN : exact-integer?  
GL\_LINE\_TOKEN : exact-integer?  
GL\_LINE\_RESET\_TOKEN : exact-integer?  
GL\_POLYGON\_TOKEN : exact-integer?  
GL\_BITMAP\_TOKEN : exact-integer?  
GL\_DRAW\_PIXEL\_TOKEN : exact-integer?  
GL\_COPY\_PIXEL\_TOKEN : exact-integer?  
GL\_PASS\_THROUGH\_TOKEN : exact-integer?  
GL\_FEEDBACK\_BUFFER\_POINTER : exact-integer?  
GL\_FEEDBACK\_BUFFER\_SIZE : exact-integer?  
GL\_FEEDBACK\_BUFFER\_TYPE : exact-integer?  
GL\_SELECTION\_BUFFER\_POINTER : exact-integer?  
GL\_SELECTION\_BUFFER\_SIZE : exact-integer?  
GL\_FOG : exact-integer?  
GL\_FOG\_MODE : exact-integer?  
GL\_FOG\_DENSITY : exact-integer?  
GL\_FOG\_COLOR : exact-integer?  
GL\_FOG\_INDEX : exact-integer?  
GL\_FOG\_START : exact-integer?  
GL\_FOG\_END : exact-integer?  
GL\_LINEAR : exact-integer?  
GL\_EXP : exact-integer?  
GL\_EXP2 : exact-integer?  
GL\_LOGIC\_OP : exact-integer?  
GL\_INDEX\_LOGIC\_OP : exact-integer?  
GL\_COLOR\_LOGIC\_OP : exact-integer?  
GL\_LOGIC\_OP\_MODE : exact-integer?

GL\_CLEAR : exact-integer?  
GL\_SET : exact-integer?  
GL\_COPY : exact-integer?  
GL\_COPY\_INVERTED : exact-integer?  
GL\_NOOP : exact-integer?  
GL\_INVERT : exact-integer?  
GL\_AND : exact-integer?  
GL\_NAND : exact-integer?  
GL\_OR : exact-integer?  
GL\_NOR : exact-integer?  
GL\_XOR : exact-integer?  
GL\_EQUIV : exact-integer?  
GL\_AND\_REVERSE : exact-integer?  
GL\_AND\_INVERTED : exact-integer?  
GL\_OR\_REVERSE : exact-integer?  
GL\_OR\_INVERTED : exact-integer?  
GL\_STENCIL\_TEST : exact-integer?  
GL\_STENCIL\_WRITEMASK : exact-integer?  
GL\_STENCIL\_BITS : exact-integer?  
GL\_STENCIL\_FUNC : exact-integer?  
GL\_STENCIL\_VALUE\_MASK : exact-integer?  
GL\_STENCIL\_REF : exact-integer?  
GL\_STENCIL\_FAIL : exact-integer?  
GL\_STENCIL\_PASS\_DEPTH\_PASS : exact-integer?  
GL\_STENCIL\_PASS\_DEPTH\_FAIL : exact-integer?  
GL\_STENCIL\_CLEAR\_VALUE : exact-integer?  
GL\_STENCIL\_INDEX : exact-integer?  
GL\_KEEP : exact-integer?  
GL\_REPLACE : exact-integer?  
GL\_INCR : exact-integer?  
GL\_DECR : exact-integer?  
GL\_NONE : exact-integer?  
GL\_LEFT : exact-integer?  
GL\_RIGHT : exact-integer?  
GL\_FRONT\_LEFT : exact-integer?  
GL\_FRONT\_RIGHT : exact-integer?  
GL\_BACK\_LEFT : exact-integer?  
GL\_BACK\_RIGHT : exact-integer?  
GL\_AUX0 : exact-integer?  
GL\_AUX1 : exact-integer?  
GL\_AUX2 : exact-integer?  
GL\_AUX3 : exact-integer?  
GL\_COLOR\_INDEX : exact-integer?  
GL\_RED : exact-integer?

GL\_GREEN : exact-integer?  
GL\_BLUE : exact-integer?  
GL\_ALPHA : exact-integer?  
GL\_LUMINANCE : exact-integer?  
GL\_LUMINANCE\_ALPHA : exact-integer?  
GL\_ALPHA\_BITS : exact-integer?  
GL\_RED\_BITS : exact-integer?  
GL\_GREEN\_BITS : exact-integer?  
GL\_BLUE\_BITS : exact-integer?  
GL\_INDEX\_BITS : exact-integer?  
GL\_SUBPIXEL\_BITS : exact-integer?  
GL\_AUX\_BUFFERS : exact-integer?  
GL\_READ\_BUFFER : exact-integer?  
GL\_DRAW\_BUFFER : exact-integer?  
GL\_DOUBLEBUFFER : exact-integer?  
GL\_STEREO : exact-integer?  
GL\_BITMAP : exact-integer?  
GL\_COLOR : exact-integer?  
GL\_DEPTH : exact-integer?  
GL\_STENCIL : exact-integer?  
GL\_DITHER : exact-integer?  
GL\_RGB : exact-integer?  
GL\_RGBA : exact-integer?  
GL\_MAX\_LIST\_NESTING : exact-integer?  
GL\_MAX\_ATTRIB\_STACK\_DEPTH : exact-integer?  
GL\_MAX\_MODELVIEW\_STACK\_DEPTH : exact-integer?  
GL\_MAX\_NAME\_STACK\_DEPTH : exact-integer?  
GL\_MAX\_PROJECTION\_STACK\_DEPTH : exact-integer?  
GL\_MAX\_TEXTURE\_STACK\_DEPTH : exact-integer?  
GL\_MAX\_EVAL\_ORDER : exact-integer?  
GL\_MAX\_LIGHTS : exact-integer?  
GL\_MAX\_CLIP\_PLANES : exact-integer?  
GL\_MAX\_TEXTURE\_SIZE : exact-integer?  
GL\_MAX\_PIXEL\_MAP\_TABLE : exact-integer?  
GL\_MAX\_VIEWPORT\_DIMS : exact-integer?  
GL\_MAX\_CLIENT\_ATTRIB\_STACK\_DEPTH : exact-integer?  
GL\_ATTRIB\_STACK\_DEPTH : exact-integer?  
GL\_CLIENT\_ATTRIB\_STACK\_DEPTH : exact-integer?  
GL\_COLOR\_CLEAR\_VALUE : exact-integer?  
GL\_COLOR\_WRITEMASK : exact-integer?  
GL\_CURRENT\_INDEX : exact-integer?  
GL\_CURRENT\_COLOR : exact-integer?  
GL\_CURRENT\_NORMAL : exact-integer?  
GL\_CURRENT\_RASTER\_COLOR : exact-integer?

GL\_CURRENT\_RASTER\_DISTANCE : exact-integer?  
GL\_CURRENT\_RASTER\_INDEX : exact-integer?  
GL\_CURRENT\_RASTER\_POSITION : exact-integer?  
GL\_CURRENT\_RASTER\_TEXTURE\_COORDS : exact-integer?  
GL\_CURRENT\_RASTER\_POSITION\_VALID : exact-integer?  
GL\_CURRENT\_TEXTURE\_COORDS : exact-integer?  
GL\_INDEX\_CLEAR\_VALUE : exact-integer?  
GL\_INDEX\_MODE : exact-integer?  
GL\_INDEX\_WRITEMASK : exact-integer?  
GL\_MODELVIEW\_MATRIX : exact-integer?  
GL\_MODELVIEW\_STACK\_DEPTH : exact-integer?  
GL\_NAME\_STACK\_DEPTH : exact-integer?  
GL\_PROJECTION\_MATRIX : exact-integer?  
GL\_PROJECTION\_STACK\_DEPTH : exact-integer?  
GL\_RENDER\_MODE : exact-integer?  
GL\_RGBA\_MODE : exact-integer?  
GL\_TEXTURE\_MATRIX : exact-integer?  
GL\_TEXTURE\_STACK\_DEPTH : exact-integer?  
GL\_VIEWPORT : exact-integer?  
GL\_AUTO\_NORMAL : exact-integer?  
GL\_MAP1\_COLOR\_4 : exact-integer?  
GL\_MAP1\_GRID\_DOMAIN : exact-integer?  
GL\_MAP1\_GRID\_SEGMENTS : exact-integer?  
GL\_MAP1\_INDEX : exact-integer?  
GL\_MAP1\_NORMAL : exact-integer?  
GL\_MAP1\_TEXTURE\_COORD\_1 : exact-integer?  
GL\_MAP1\_TEXTURE\_COORD\_2 : exact-integer?  
GL\_MAP1\_TEXTURE\_COORD\_3 : exact-integer?  
GL\_MAP1\_TEXTURE\_COORD\_4 : exact-integer?  
GL\_MAP1\_VERTEX\_3 : exact-integer?  
GL\_MAP1\_VERTEX\_4 : exact-integer?  
GL\_MAP2\_COLOR\_4 : exact-integer?  
GL\_MAP2\_GRID\_DOMAIN : exact-integer?  
GL\_MAP2\_GRID\_SEGMENTS : exact-integer?  
GL\_MAP2\_INDEX : exact-integer?  
GL\_MAP2\_NORMAL : exact-integer?  
GL\_MAP2\_TEXTURE\_COORD\_1 : exact-integer?  
GL\_MAP2\_TEXTURE\_COORD\_2 : exact-integer?  
GL\_MAP2\_TEXTURE\_COORD\_3 : exact-integer?  
GL\_MAP2\_TEXTURE\_COORD\_4 : exact-integer?  
GL\_MAP2\_VERTEX\_3 : exact-integer?  
GL\_MAP2\_VERTEX\_4 : exact-integer?  
GL\_COEFF : exact-integer?  
GL\_DOMAIN : exact-integer?



GL\_ORDER : exact-integer?  
GL\_FOG\_HINT : exact-integer?  
GL\_LINE\_SMOOTH\_HINT : exact-integer?  
GL\_PERSPECTIVE\_CORRECTION\_HINT : exact-integer?  
GL\_POINT\_SMOOTH\_HINT : exact-integer?  
GL\_POLYGON\_SMOOTH\_HINT : exact-integer?  
GL\_DONT\_CARE : exact-integer?  
GL\_FASTEST : exact-integer?  
GL\_NICEST : exact-integer?  
GL\_SCISSOR\_TEST : exact-integer?  
GL\_SCISSOR\_BOX : exact-integer?  
GL\_MAP\_COLOR : exact-integer?  
GL\_MAP\_STENCIL : exact-integer?  
GL\_INDEX\_SHIFT : exact-integer?  
GL\_INDEX\_OFFSET : exact-integer?  
GL\_RED\_SCALE : exact-integer?  
GL\_RED\_BIAS : exact-integer?  
GL\_GREEN\_SCALE : exact-integer?  
GL\_GREEN\_BIAS : exact-integer?  
GL\_BLUE\_SCALE : exact-integer?  
GL\_BLUE\_BIAS : exact-integer?  
GL\_ALPHA\_SCALE : exact-integer?  
GL\_ALPHA\_BIAS : exact-integer?  
GL\_DEPTH\_SCALE : exact-integer?  
GL\_DEPTH\_BIAS : exact-integer?  
GL\_PIXEL\_MAP\_S\_TO\_S\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_I\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_R\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_G\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_B\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_A\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_R\_TO\_R\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_G\_TO\_G\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_B\_TO\_B\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_A\_TO\_A\_SIZE : exact-integer?  
GL\_PIXEL\_MAP\_S\_TO\_S : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_I : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_R : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_G : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_B : exact-integer?  
GL\_PIXEL\_MAP\_I\_TO\_A : exact-integer?  
GL\_PIXEL\_MAP\_R\_TO\_R : exact-integer?  
GL\_PIXEL\_MAP\_G\_TO\_G : exact-integer?  
GL\_PIXEL\_MAP\_B\_TO\_B : exact-integer?

GL\_PIXEL\_MAP\_A\_TO\_A : exact-integer?  
GL\_PACK\_ALIGNMENT : exact-integer?  
GL\_PACK\_LSB\_FIRST : exact-integer?  
GL\_PACK\_ROW\_LENGTH : exact-integer?  
GL\_PACK\_SKIP\_PIXELS : exact-integer?  
GL\_PACK\_SKIP\_ROWS : exact-integer?  
GL\_PACK\_SWAP\_BYTES : exact-integer?  
GL\_UNPACK\_ALIGNMENT : exact-integer?  
GL\_UNPACK\_LSB\_FIRST : exact-integer?  
GL\_UNPACK\_ROW\_LENGTH : exact-integer?  
GL\_UNPACK\_SKIP\_PIXELS : exact-integer?  
GL\_UNPACK\_SKIP\_ROWS : exact-integer?  
GL\_UNPACK\_SWAP\_BYTES : exact-integer?  
GL\_ZOOM\_X : exact-integer?  
GL\_ZOOM\_Y : exact-integer?  
GL\_TEXTURE\_ENV : exact-integer?  
GL\_TEXTURE\_ENV\_MODE : exact-integer?  
GL\_TEXTURE\_1D : exact-integer?  
GL\_TEXTURE\_2D : exact-integer?  
GL\_TEXTURE\_WRAP\_S : exact-integer?  
GL\_TEXTURE\_WRAP\_T : exact-integer?  
GL\_TEXTURE\_MAG\_FILTER : exact-integer?  
GL\_TEXTURE\_MIN\_FILTER : exact-integer?  
GL\_TEXTURE\_ENV\_COLOR : exact-integer?  
GL\_TEXTURE\_GEN\_S : exact-integer?  
GL\_TEXTURE\_GEN\_T : exact-integer?  
GL\_TEXTURE\_GEN\_MODE : exact-integer?  
GL\_TEXTURE\_BORDER\_COLOR : exact-integer?  
GL\_TEXTURE\_WIDTH : exact-integer?  
GL\_TEXTURE\_HEIGHT : exact-integer?  
GL\_TEXTURE\_BORDER : exact-integer?  
GL\_TEXTURE\_COMPONENTS : exact-integer?  
GL\_TEXTURE\_RED\_SIZE : exact-integer?  
GL\_TEXTURE\_GREEN\_SIZE : exact-integer?  
GL\_TEXTURE\_BLUE\_SIZE : exact-integer?  
GL\_TEXTURE\_ALPHA\_SIZE : exact-integer?  
GL\_TEXTURE\_LUMINANCE\_SIZE : exact-integer?  
GL\_TEXTURE\_INTENSITY\_SIZE : exact-integer?  
GL\_NEAREST\_MIPMAP\_NEAREST : exact-integer?  
GL\_NEAREST\_MIPMAP\_LINEAR : exact-integer?  
GL\_LINEAR\_MIPMAP\_NEAREST : exact-integer?  
GL\_LINEAR\_MIPMAP\_LINEAR : exact-integer?  
GL\_OBJECT\_LINEAR : exact-integer?  
GL\_OBJECT\_PLANE : exact-integer?

GL\_EYE\_LINEAR : exact-integer?  
GL\_EYE\_PLANE : exact-integer?  
GL\_SPHERE\_MAP : exact-integer?  
GL\_DECAL : exact-integer?  
GL\_MODULATE : exact-integer?  
GL\_NEAREST : exact-integer?  
GL\_REPEAT : exact-integer?  
GL\_CLAMP : exact-integer?  
GL\_S : exact-integer?  
GL\_T : exact-integer?  
GL\_R : exact-integer?  
GL\_Q : exact-integer?  
GL\_TEXTURE\_GEN\_R : exact-integer?  
GL\_TEXTURE\_GEN\_Q : exact-integer?  
GL\_VENDOR : exact-integer?  
GL\_RENDERER : exact-integer?  
GL\_VERSION : exact-integer?  
GL\_EXTENSIONS : exact-integer?  
GL\_NO\_ERROR : exact-integer?  
GL\_INVALID\_VALUE : exact-integer?  
GL\_INVALID\_ENUM : exact-integer?  
GL\_INVALID\_OPERATION : exact-integer?  
GL\_STACK\_OVERFLOW : exact-integer?  
GL\_STACK\_UNDERFLOW : exact-integer?  
GL\_OUT\_OF\_MEMORY : exact-integer?  
GL\_CURRENT\_BIT : exact-integer?  
GL\_POINT\_BIT : exact-integer?  
GL\_LINE\_BIT : exact-integer?  
GL\_POLYGON\_BIT : exact-integer?  
GL\_POLYGON\_STIPPLE\_BIT : exact-integer?  
GL\_PIXEL\_MODE\_BIT : exact-integer?  
GL\_LIGHTING\_BIT : exact-integer?  
GL\_FOG\_BIT : exact-integer?  
GL\_DEPTH\_BUFFER\_BIT : exact-integer?  
GL\_ACCUM\_BUFFER\_BIT : exact-integer?  
GL\_STENCIL\_BUFFER\_BIT : exact-integer?  
GL\_VIEWPORT\_BIT : exact-integer?  
GL\_TRANSFORM\_BIT : exact-integer?  
GL\_ENABLE\_BIT : exact-integer?  
GL\_COLOR\_BUFFER\_BIT : exact-integer?  
GL\_HINT\_BIT : exact-integer?  
GL\_EVAL\_BIT : exact-integer?  
GL\_LIST\_BIT : exact-integer?  
GL\_TEXTURE\_BIT : exact-integer?

GL\_SCISSOR\_BIT : exact-integer?  
GL\_ALL\_ATTRIB\_BITS : exact-integer?  
GL\_PROXY\_TEXTURE\_1D : exact-integer?  
GL\_PROXY\_TEXTURE\_2D : exact-integer?  
GL\_TEXTURE\_PRIORITY : exact-integer?  
GL\_TEXTURE\_RESIDENT : exact-integer?  
GL\_TEXTURE\_BINDING\_1D : exact-integer?  
GL\_TEXTURE\_BINDING\_2D : exact-integer?  
GL\_TEXTURE\_INTERNAL\_FORMAT : exact-integer?  
GL\_ALPHA4 : exact-integer?  
GL\_ALPHA8 : exact-integer?  
GL\_ALPHA12 : exact-integer?  
GL\_ALPHA16 : exact-integer?  
GL\_LUMINANCE4 : exact-integer?  
GL\_LUMINANCE8 : exact-integer?  
GL\_LUMINANCE12 : exact-integer?  
GL\_LUMINANCE16 : exact-integer?  
GL\_LUMINANCE4\_ALPHA4 : exact-integer?  
GL\_LUMINANCE6\_ALPHA2 : exact-integer?  
GL\_LUMINANCE8\_ALPHA8 : exact-integer?  
GL\_LUMINANCE12\_ALPHA4 : exact-integer?  
GL\_LUMINANCE12\_ALPHA12 : exact-integer?  
GL\_LUMINANCE16\_ALPHA16 : exact-integer?  
GL\_INTENSITY : exact-integer?  
GL\_INTENSITY4 : exact-integer?  
GL\_INTENSITY8 : exact-integer?  
GL\_INTENSITY12 : exact-integer?  
GL\_INTENSITY16 : exact-integer?  
GL\_R3\_G3\_B2 : exact-integer?  
GL\_RGB4 : exact-integer?  
GL\_RGB5 : exact-integer?  
GL\_RGB8 : exact-integer?  
GL\_RGB10 : exact-integer?  
GL\_RGB12 : exact-integer?  
GL\_RGB16 : exact-integer?  
GL\_RGBA2 : exact-integer?  
GL\_RGBA4 : exact-integer?  
GL\_RGB5\_A1 : exact-integer?  
GL\_RGBA8 : exact-integer?  
GL\_RGB10\_A2 : exact-integer?  
GL\_RGBA12 : exact-integer?  
GL\_RGBA16 : exact-integer?  
GL\_CLIENT\_PIXEL\_STORE\_BIT : exact-integer?  
GL\_CLIENT\_VERTEX\_ARRAY\_BIT : exact-integer?

GL\_ALL\_CLIENT\_ATTRIB\_BITS : exact-integer?  
GL\_CLIENT\_ALL\_ATTRIB\_BITS : exact-integer?  
GL\_UNSIGNED\_BYTE\_3\_3\_2 : exact-integer?  
GL\_UNSIGNED\_SHORT\_4\_4\_4\_4 : exact-integer?  
GL\_UNSIGNED\_SHORT\_5\_5\_5\_1 : exact-integer?  
GL\_UNSIGNED\_INT\_8\_8\_8\_8 : exact-integer?  
GL\_UNSIGNED\_INT\_10\_10\_10\_2 : exact-integer?  
GL\_RESCALE\_NORMAL : exact-integer?  
GL\_TEXTURE\_BINDING\_3D : exact-integer?  
GL\_PACK\_SKIP\_IMAGES : exact-integer?  
GL\_PACK\_IMAGE\_HEIGHT : exact-integer?  
GL\_UNPACK\_SKIP\_IMAGES : exact-integer?  
GL\_UNPACK\_IMAGE\_HEIGHT : exact-integer?  
GL\_TEXTURE\_3D : exact-integer?  
GL\_PROXY\_TEXTURE\_3D : exact-integer?  
GL\_TEXTURE\_DEPTH : exact-integer?  
GL\_TEXTURE\_WRAP\_R : exact-integer?  
GL\_MAX\_3D\_TEXTURE\_SIZE : exact-integer?  
GL\_UNSIGNED\_BYTE\_2\_3\_3\_REV : exact-integer?  
GL\_UNSIGNED\_SHORT\_5\_6\_5 : exact-integer?  
GL\_UNSIGNED\_SHORT\_5\_6\_5\_REV : exact-integer?  
GL\_UNSIGNED\_SHORT\_4\_4\_4\_4\_REV : exact-integer?  
GL\_UNSIGNED\_SHORT\_1\_5\_5\_5\_REV : exact-integer?  
GL\_UNSIGNED\_INT\_8\_8\_8\_8\_REV : exact-integer?  
GL\_UNSIGNED\_INT\_2\_10\_10\_10\_REV : exact-integer?  
GL\_BGR : exact-integer?  
GL\_BGRA : exact-integer?  
GL\_MAX\_ELEMENTS\_VERTICES : exact-integer?  
GL\_MAX\_ELEMENTS\_INDICES : exact-integer?  
GL\_CLAMP\_TO\_EDGE : exact-integer?  
GL\_TEXTURE\_MIN\_LOD : exact-integer?  
GL\_TEXTURE\_MAX\_LOD : exact-integer?  
GL\_TEXTURE\_BASE\_LEVEL : exact-integer?  
GL\_TEXTURE\_MAX\_LEVEL : exact-integer?  
GL\_LIGHT\_MODEL\_COLOR\_CONTROL : exact-integer?  
GL\_SINGLE\_COLOR : exact-integer?  
GL\_SEPARATE\_SPECULAR\_COLOR : exact-integer?  
GL\_SMOOTH\_POINT\_SIZE\_RANGE : exact-integer?  
GL\_SMOOTH\_POINT\_SIZE\_GRANULARITY : exact-integer?  
GL\_SMOOTH\_LINE\_WIDTH\_RANGE : exact-integer?  
GL\_SMOOTH\_LINE\_WIDTH\_GRANULARITY : exact-integer?  
GL\_ALIASED\_POINT\_SIZE\_RANGE : exact-integer?  
GL\_ALIASED\_LINE\_WIDTH\_RANGE : exact-integer?  
GL\_CONSTANT\_COLOR : exact-integer?

GL\_ONE\_MINUS\_CONSTANT\_COLOR : exact-integer?  
GL\_CONSTANT\_ALPHA : exact-integer?  
GL\_ONE\_MINUS\_CONSTANT\_ALPHA : exact-integer?  
GL\_BLEND\_COLOR : exact-integer?  
GL\_FUNC\_ADD : exact-integer?  
GL\_MIN : exact-integer?  
GL\_MAX : exact-integer?  
GL\_BLEND\_EQUATION : exact-integer?  
GL\_FUNC\_SUBTRACT : exact-integer?  
GL\_FUNC\_REVERSE\_SUBTRACT : exact-integer?  
GL\_CONVOLUTION\_1D : exact-integer?  
GL\_CONVOLUTION\_2D : exact-integer?  
GL\_SEPARABLE\_2D : exact-integer?  
GL\_CONVOLUTION\_BORDER\_MODE : exact-integer?  
GL\_CONVOLUTION\_FILTER\_SCALE : exact-integer?  
GL\_CONVOLUTION\_FILTER\_BIAS : exact-integer?  
GL\_REDUCE : exact-integer?  
GL\_CONVOLUTION\_FORMAT : exact-integer?  
GL\_CONVOLUTION\_WIDTH : exact-integer?  
GL\_CONVOLUTION\_HEIGHT : exact-integer?  
GL\_MAX\_CONVOLUTION\_WIDTH : exact-integer?  
GL\_MAX\_CONVOLUTION\_HEIGHT : exact-integer?  
GL\_POST\_CONVOLUTION\_RED\_SCALE : exact-integer?  
GL\_POST\_CONVOLUTION\_GREEN\_SCALE : exact-integer?  
GL\_POST\_CONVOLUTION\_BLUE\_SCALE : exact-integer?  
GL\_POST\_CONVOLUTION\_ALPHA\_SCALE : exact-integer?  
GL\_POST\_CONVOLUTION\_RED\_BIAS : exact-integer?  
GL\_POST\_CONVOLUTION\_GREEN\_BIAS : exact-integer?  
GL\_POST\_CONVOLUTION\_BLUE\_BIAS : exact-integer?  
GL\_POST\_CONVOLUTION\_ALPHA\_BIAS : exact-integer?  
GL\_HISTOGRAM : exact-integer?  
GL\_PROXY\_HISTOGRAM : exact-integer?  
GL\_HISTOGRAM\_WIDTH : exact-integer?  
GL\_HISTOGRAM\_FORMAT : exact-integer?  
GL\_HISTOGRAM\_RED\_SIZE : exact-integer?  
GL\_HISTOGRAM\_GREEN\_SIZE : exact-integer?  
GL\_HISTOGRAM\_BLUE\_SIZE : exact-integer?  
GL\_HISTOGRAM\_ALPHA\_SIZE : exact-integer?  
GL\_HISTOGRAM\_LUMINANCE\_SIZE : exact-integer?  
GL\_HISTOGRAM\_SINK : exact-integer?  
GL\_MINMAX : exact-integer?  
GL\_MINMAX\_FORMAT : exact-integer?  
GL\_MINMAX\_SINK : exact-integer?  
GL\_TABLE\_TOO\_LARGE : exact-integer?

GL\_COLOR\_MATRIX : exact-integer?  
GL\_COLOR\_MATRIX\_STACK\_DEPTH : exact-integer?  
GL\_MAX\_COLOR\_MATRIX\_STACK\_DEPTH : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_RED\_SCALE : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_GREEN\_SCALE : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_BLUE\_SCALE : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_ALPHA\_SCALE : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_RED\_BIAS : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_GREEN\_BIAS : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_BLUE\_BIAS : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_ALPHA\_BIAS : exact-integer?  
GL\_COLOR\_TABLE : exact-integer?  
GL\_POST\_CONVOLUTION\_COLOR\_TABLE : exact-integer?  
GL\_POST\_COLOR\_MATRIX\_COLOR\_TABLE : exact-integer?  
GL\_PROXY\_COLOR\_TABLE : exact-integer?  
GL\_PROXY\_POST\_CONVOLUTION\_COLOR\_TABLE : exact-integer?  
GL\_PROXY\_POST\_COLOR\_MATRIX\_COLOR\_TABLE : exact-integer?  
GL\_COLOR\_TABLE\_SCALE : exact-integer?  
GL\_COLOR\_TABLE\_BIAS : exact-integer?  
GL\_COLOR\_TABLE\_FORMAT : exact-integer?  
GL\_COLOR\_TABLE\_WIDTH : exact-integer?  
GL\_COLOR\_TABLE\_RED\_SIZE : exact-integer?  
GL\_COLOR\_TABLE\_GREEN\_SIZE : exact-integer?  
GL\_COLOR\_TABLE\_BLUE\_SIZE : exact-integer?  
GL\_COLOR\_TABLE\_ALPHA\_SIZE : exact-integer?  
GL\_COLOR\_TABLE\_LUMINANCE\_SIZE : exact-integer?  
GL\_COLOR\_TABLE\_INTENSITY\_SIZE : exact-integer?  
GL\_CONSTANT\_BORDER : exact-integer?  
GL\_REPLICATE\_BORDER : exact-integer?  
GL\_CONVOLUTION\_BORDER\_COLOR : exact-integer?  
GL\_TEXTURE0 : exact-integer?  
GL\_TEXTURE1 : exact-integer?  
GL\_TEXTURE2 : exact-integer?  
GL\_TEXTURE3 : exact-integer?  
GL\_TEXTURE4 : exact-integer?  
GL\_TEXTURE5 : exact-integer?  
GL\_TEXTURE6 : exact-integer?  
GL\_TEXTURE7 : exact-integer?  
GL\_TEXTURE8 : exact-integer?  
GL\_TEXTURE9 : exact-integer?  
GL\_TEXTURE10 : exact-integer?  
GL\_TEXTURE11 : exact-integer?  
GL\_TEXTURE12 : exact-integer?  
GL\_TEXTURE13 : exact-integer?

GL\_TEXTURE14 : exact-integer?  
GL\_TEXTURE15 : exact-integer?  
GL\_TEXTURE16 : exact-integer?  
GL\_TEXTURE17 : exact-integer?  
GL\_TEXTURE18 : exact-integer?  
GL\_TEXTURE19 : exact-integer?  
GL\_TEXTURE20 : exact-integer?  
GL\_TEXTURE21 : exact-integer?  
GL\_TEXTURE22 : exact-integer?  
GL\_TEXTURE23 : exact-integer?  
GL\_TEXTURE24 : exact-integer?  
GL\_TEXTURE25 : exact-integer?  
GL\_TEXTURE26 : exact-integer?  
GL\_TEXTURE27 : exact-integer?  
GL\_TEXTURE28 : exact-integer?  
GL\_TEXTURE29 : exact-integer?  
GL\_TEXTURE30 : exact-integer?  
GL\_TEXTURE31 : exact-integer?  
GL\_ACTIVE\_TEXTURE : exact-integer?  
GL\_CLIENT\_ACTIVE\_TEXTURE : exact-integer?  
GL\_MAX\_TEXTURE\_UNITS : exact-integer?  
GL\_TRANSPOSE\_MODELVIEW\_MATRIX : exact-integer?  
GL\_TRANSPOSE\_PROJECTION\_MATRIX : exact-integer?  
GL\_TRANSPOSE\_TEXTURE\_MATRIX : exact-integer?  
GL\_TRANSPOSE\_COLOR\_MATRIX : exact-integer?  
GL\_MULTISAMPLE : exact-integer?  
GL\_SAMPLE\_ALPHA\_TO\_COVERAGE : exact-integer?  
GL\_SAMPLE\_ALPHA\_TO\_ONE : exact-integer?  
GL\_SAMPLE\_COVERAGE : exact-integer?  
GL\_SAMPLE\_BUFFERS : exact-integer?  
GL\_SAMPLES : exact-integer?  
GL\_SAMPLE\_COVERAGE\_VALUE : exact-integer?  
GL\_SAMPLE\_COVERAGE\_INVERT : exact-integer?  
GL\_MULTISAMPLE\_BIT : exact-integer?  
GL\_NORMAL\_MAP : exact-integer?  
GL\_REFLECTION\_MAP : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP : exact-integer?  
GL\_TEXTURE\_BINDING\_CUBE\_MAP : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP\_POSITIVE\_X : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP\_NEGATIVE\_X : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP\_POSITIVE\_Y : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP\_NEGATIVE\_Y : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP\_POSITIVE\_Z : exact-integer?  
GL\_TEXTURE\_CUBE\_MAP\_NEGATIVE\_Z : exact-integer?



GL\_PROXY\_TEXTURE\_CUBE\_MAP : exact-integer?  
GL\_MAX\_CUBE\_MAP\_TEXTURE\_SIZE : exact-integer?  
GL\_COMPRESSED\_ALPHA : exact-integer?  
GL\_COMPRESSED\_LUMINANCE : exact-integer?  
GL\_COMPRESSED\_LUMINANCE\_ALPHA : exact-integer?  
GL\_COMPRESSED\_INTENSITY : exact-integer?  
GL\_COMPRESSED\_RGB : exact-integer?  
GL\_COMPRESSED\_RGBA : exact-integer?  
GL\_TEXTURE\_COMPRESSION\_HINT : exact-integer?  
GL\_TEXTURE\_COMPRESSED\_IMAGE\_SIZE : exact-integer?  
GL\_TEXTURE\_COMPRESSED : exact-integer?  
GL\_NUM\_COMPRESSED\_TEXTURE\_FORMATS : exact-integer?  
GL\_COMPRESSED\_TEXTURE\_FORMATS : exact-integer?  
GL\_CLAMP\_TO\_BORDER : exact-integer?  
GL\_COMBINE : exact-integer?  
GL\_COMBINE\_RGB : exact-integer?  
GL\_COMBINE\_ALPHA : exact-integer?  
GL\_SOURCE0\_RGB : exact-integer?  
GL\_SOURCE1\_RGB : exact-integer?  
GL\_SOURCE2\_RGB : exact-integer?  
GL\_SOURCE0\_ALPHA : exact-integer?  
GL\_SOURCE1\_ALPHA : exact-integer?  
GL\_SOURCE2\_ALPHA : exact-integer?  
GL\_OPERAND0\_RGB : exact-integer?  
GL\_OPERAND1\_RGB : exact-integer?  
GL\_OPERAND2\_RGB : exact-integer?  
GL\_OPERAND0\_ALPHA : exact-integer?  
GL\_OPERAND1\_ALPHA : exact-integer?  
GL\_OPERAND2\_ALPHA : exact-integer?  
GL\_RGB\_SCALE : exact-integer?  
GL\_ADD\_SIGNED : exact-integer?  
GL\_INTERPOLATE : exact-integer?  
GL\_SUBTRACT : exact-integer?  
GL\_CONSTANT : exact-integer?  
GL\_PRIMARY\_COLOR : exact-integer?  
GL\_PREVIOUS : exact-integer?  
GL\_DOT3\_RGB : exact-integer?  
GL\_DOT3\_RGBA : exact-integer?  
GL\_BLEND\_DST\_RGB : exact-integer?  
GL\_BLEND\_SRC\_RGB : exact-integer?  
GL\_BLEND\_DST\_ALPHA : exact-integer?  
GL\_BLEND\_SRC\_ALPHA : exact-integer?  
GL\_POINT\_SIZE\_MIN : exact-integer?  
GL\_POINT\_SIZE\_MAX : exact-integer?

GL\_POINT\_FADE\_THRESHOLD\_SIZE : exact-integer?  
GL\_POINT\_DISTANCE\_ATTENUATION : exact-integer?  
GL\_GENERATE\_MIPMAP : exact-integer?  
GL\_GENERATE\_MIPMAP\_HINT : exact-integer?  
GL\_DEPTH\_COMPONENT16 : exact-integer?  
GL\_DEPTH\_COMPONENT24 : exact-integer?  
GL\_DEPTH\_COMPONENT32 : exact-integer?  
GL\_MIRRORED\_REPEAT : exact-integer?  
GL\_FOG\_COORDINATE\_SOURCE : exact-integer?  
GL\_FOG\_COORDINATE : exact-integer?  
GL\_FRAGMENT\_DEPTH : exact-integer?  
GL\_CURRENT\_FOG\_COORDINATE : exact-integer?  
GL\_FOG\_COORDINATE\_ARRAY\_TYPE : exact-integer?  
GL\_FOG\_COORDINATE\_ARRAY\_STRIDE : exact-integer?  
GL\_FOG\_COORDINATE\_ARRAY\_POINTER : exact-integer?  
GL\_FOG\_COORDINATE\_ARRAY : exact-integer?  
GL\_COLOR\_SUM : exact-integer?  
GL\_CURRENT\_SECONDARY\_COLOR : exact-integer?  
GL\_SECONDARY\_COLOR\_ARRAY\_SIZE : exact-integer?  
GL\_SECONDARY\_COLOR\_ARRAY\_TYPE : exact-integer?  
GL\_SECONDARY\_COLOR\_ARRAY\_STRIDE : exact-integer?  
GL\_SECONDARY\_COLOR\_ARRAY\_POINTER : exact-integer?  
GL\_SECONDARY\_COLOR\_ARRAY : exact-integer?  
GL\_MAX\_TEXTURE\_LOD\_BIAS : exact-integer?  
GL\_TEXTURE\_FILTER\_CONTROL : exact-integer?  
GL\_TEXTURE\_LOD\_BIAS : exact-integer?  
GL\_INCR\_WRAP : exact-integer?  
GL DECR\_WRAP : exact-integer?  
GL\_TEXTURE\_DEPTH\_SIZE : exact-integer?  
GL\_DEPTH\_TEXTURE\_MODE : exact-integer?  
GL\_TEXTURE\_COMPARE\_MODE : exact-integer?  
GL\_TEXTURE\_COMPARE\_FUNC : exact-integer?  
GL\_COMPARE\_R\_TO\_TEXTURE : exact-integer?  
GL\_BUFFER\_SIZE : exact-integer?  
GL\_BUFFER\_USAGE : exact-integer?  
GL\_QUERY\_COUNTER\_BITS : exact-integer?  
GL\_CURRENT\_QUERY : exact-integer?  
GL\_QUERY\_RESULT : exact-integer?  
GL\_QUERY\_RESULT\_AVAILABLE : exact-integer?  
GL\_ARRAY\_BUFFER : exact-integer?  
GL\_ELEMENT\_ARRAY\_BUFFER : exact-integer?  
GL\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_ELEMENT\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_VERTEX\_ARRAY\_BUFFER\_BINDING : exact-integer?

GL\_NORMAL\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_COLOR\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_INDEX\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_TEXTURE\_COORD\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_EDGE\_FLAG\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_SECONDARY\_COLOR\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_FOG\_COORDINATE\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_WEIGHT\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_VERTEX\_ATTRIB\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_READ\_ONLY : exact-integer?  
GL\_WRITE\_ONLY : exact-integer?  
GL\_READ\_WRITE : exact-integer?  
GL\_BUFFER\_ACCESS : exact-integer?  
GL\_BUFFER\_MAPPED : exact-integer?  
GL\_BUFFER\_MAP\_POINTER : exact-integer?  
GL\_STREAM\_DRAW : exact-integer?  
GL\_STREAM\_READ : exact-integer?  
GL\_STREAM\_COPY : exact-integer?  
GL\_STATIC\_DRAW : exact-integer?  
GL\_STATIC\_READ : exact-integer?  
GL\_STATIC\_COPY : exact-integer?  
GL\_DYNAMIC\_DRAW : exact-integer?  
GL\_DYNAMIC\_READ : exact-integer?  
GL\_DYNAMIC\_COPY : exact-integer?  
GL\_SAMPLES\_PASSED : exact-integer?  
GL\_FOG\_COORD\_SRC : exact-integer?  
GL\_FOG\_COORD : exact-integer?  
GL\_CURRENT\_FOG\_COORD : exact-integer?  
GL\_FOG\_COORD\_ARRAY\_TYPE : exact-integer?  
GL\_FOG\_COORD\_ARRAY\_STRIDE : exact-integer?  
GL\_FOG\_COORD\_ARRAY\_POINTER : exact-integer?  
GL\_FOG\_COORD\_ARRAY : exact-integer?  
GL\_FOG\_COORD\_ARRAY\_BUFFER\_BINDING : exact-integer?  
GL\_SRC0\_RGB : exact-integer?  
GL\_SRC1\_RGB : exact-integer?  
GL\_SRC2\_RGB : exact-integer?  
GL\_SRC0\_ALPHA : exact-integer?  
GL\_SRC1\_ALPHA : exact-integer?  
GL\_SRC2\_ALPHA : exact-integer?  
GLU\_FALSE : exact-integer?  
GLU\_TRUE : exact-integer?  
GLU\_VERSION : exact-integer?  
GLU\_EXTENSIONS : exact-integer?  
GLU\_INVALID\_ENUM : exact-integer?

GLU\_INVALID\_VALUE : exact-integer?  
GLU\_OUT\_OF\_MEMORY : exact-integer?  
GLU\_INVALID\_OPERATION : exact-integer?  
GLU\_OUTLINE\_POLYGON : exact-integer?  
GLU\_OUTLINE\_PATCH : exact-integer?  
GLU\_NURBS\_ERROR : exact-integer?  
GLU\_ERROR : exact-integer?  
GLU\_NURBS\_BEGIN : exact-integer?  
GLU\_NURBS\_BEGIN\_EXT : exact-integer?  
GLU\_NURBS\_VERTEX : exact-integer?  
GLU\_NURBS\_VERTEX\_EXT : exact-integer?  
GLU\_NURBS\_NORMAL : exact-integer?  
GLU\_NURBS\_NORMAL\_EXT : exact-integer?  
GLU\_NURBS\_COLOR : exact-integer?  
GLU\_NURBS\_COLOR\_EXT : exact-integer?  
GLU\_NURBS\_TEXTURE\_COORD : exact-integer?  
GLU\_NURBS\_TEX\_COORD\_EXT : exact-integer?  
GLU\_NURBS\_END : exact-integer?  
GLU\_NURBS\_END\_EXT : exact-integer?  
GLU\_NURBS\_BEGIN\_DATA : exact-integer?  
GLU\_NURBS\_BEGIN\_DATA\_EXT : exact-integer?  
GLU\_NURBS\_VERTEX\_DATA : exact-integer?  
GLU\_NURBS\_VERTEX\_DATA\_EXT : exact-integer?  
GLU\_NURBS\_NORMAL\_DATA : exact-integer?  
GLU\_NURBS\_NORMAL\_DATA\_EXT : exact-integer?  
GLU\_NURBS\_COLOR\_DATA : exact-integer?  
GLU\_NURBS\_COLOR\_DATA\_EXT : exact-integer?  
GLU\_NURBS\_TEXTURE\_COORD\_DATA : exact-integer?  
GLU\_NURBS\_TEX\_COORD\_DATA\_EXT : exact-integer?  
GLU\_NURBS\_END\_DATA : exact-integer?  
GLU\_NURBS\_END\_DATA\_EXT : exact-integer?  
GLU\_NURBS\_ERROR1 : exact-integer?  
GLU\_NURBS\_ERROR2 : exact-integer?  
GLU\_NURBS\_ERROR3 : exact-integer?  
GLU\_NURBS\_ERROR4 : exact-integer?  
GLU\_NURBS\_ERROR5 : exact-integer?  
GLU\_NURBS\_ERROR6 : exact-integer?  
GLU\_NURBS\_ERROR7 : exact-integer?  
GLU\_NURBS\_ERROR8 : exact-integer?  
GLU\_NURBS\_ERROR9 : exact-integer?  
GLU\_NURBS\_ERROR10 : exact-integer?  
GLU\_NURBS\_ERROR11 : exact-integer?  
GLU\_NURBS\_ERROR12 : exact-integer?  
GLU\_NURBS\_ERROR13 : exact-integer?

GLU\_NURBS\_ERROR14 : exact-integer?  
GLU\_NURBS\_ERROR15 : exact-integer?  
GLU\_NURBS\_ERROR16 : exact-integer?  
GLU\_NURBS\_ERROR17 : exact-integer?  
GLU\_NURBS\_ERROR18 : exact-integer?  
GLU\_NURBS\_ERROR19 : exact-integer?  
GLU\_NURBS\_ERROR20 : exact-integer?  
GLU\_NURBS\_ERROR21 : exact-integer?  
GLU\_NURBS\_ERROR22 : exact-integer?  
GLU\_NURBS\_ERROR23 : exact-integer?  
GLU\_NURBS\_ERROR24 : exact-integer?  
GLU\_NURBS\_ERROR25 : exact-integer?  
GLU\_NURBS\_ERROR26 : exact-integer?  
GLU\_NURBS\_ERROR27 : exact-integer?  
GLU\_NURBS\_ERROR28 : exact-integer?  
GLU\_NURBS\_ERROR29 : exact-integer?  
GLU\_NURBS\_ERROR30 : exact-integer?  
GLU\_NURBS\_ERROR31 : exact-integer?  
GLU\_NURBS\_ERROR32 : exact-integer?  
GLU\_NURBS\_ERROR33 : exact-integer?  
GLU\_NURBS\_ERROR34 : exact-integer?  
GLU\_NURBS\_ERROR35 : exact-integer?  
GLU\_NURBS\_ERROR36 : exact-integer?  
GLU\_NURBS\_ERROR37 : exact-integer?  
GLU\_AUTO\_LOAD\_MATRIX : exact-integer?  
GLU\_CULLING : exact-integer?  
GLU\_SAMPLING\_TOLERANCE : exact-integer?  
GLU\_DISPLAY\_MODE : exact-integer?  
GLU\_PARAMETRIC\_TOLERANCE : exact-integer?  
GLU\_SAMPLING\_METHOD : exact-integer?  
GLU\_U\_STEP : exact-integer?  
GLU\_V\_STEP : exact-integer?  
GLU\_NURBS\_MODE : exact-integer?  
GLU\_NURBS\_MODE\_EXT : exact-integer?  
GLU\_NURBS\_TESSELLATOR : exact-integer?  
GLU\_NURBS\_TESSELLATOR\_EXT : exact-integer?  
GLU\_NURBS\_RENDERER : exact-integer?  
GLU\_NURBS\_RENDERER\_EXT : exact-integer?  
GLU\_OBJECT\_PARAMETRIC\_ERROR : exact-integer?  
GLU\_OBJECT\_PARAMETRIC\_ERROR\_EXT : exact-integer?  
GLU\_OBJECT\_PATH\_LENGTH : exact-integer?  
GLU\_OBJECT\_PATH\_LENGTH\_EXT : exact-integer?  
GLU\_PATH\_LENGTH : exact-integer?  
GLU\_PARAMETRIC\_ERROR : exact-integer?

GLU\_DOMAIN\_DISTANCE : exact-integer?  
GLU\_MAP1\_TRIM\_2 : exact-integer?  
GLU\_MAP1\_TRIM\_3 : exact-integer?  
GLU\_POINT : exact-integer?  
GLU\_LINE : exact-integer?  
GLU\_FILL : exact-integer?  
GLU\_SILHOUETTE : exact-integer?  
GLU\_SMOOTH : exact-integer?  
GLU\_FLAT : exact-integer?  
GLU\_NONE : exact-integer?  
GLU\_OUTSIDE : exact-integer?  
GLU\_INSIDE : exact-integer?  
GLU\_TESS\_BEGIN : exact-integer?  
GLU\_BEGIN : exact-integer?  
GLU\_TESS\_VERTEX : exact-integer?  
GLU\_VERTEX : exact-integer?  
GLU\_TESS\_END : exact-integer?  
GLU\_END : exact-integer?  
GLU\_TESS\_ERROR : exact-integer?  
GLU\_TESS\_EDGE\_FLAG : exact-integer?  
GLU\_EDGE\_FLAG : exact-integer?  
GLU\_TESS\_COMBINE : exact-integer?  
GLU\_TESS\_BEGIN\_DATA : exact-integer?  
GLU\_TESS\_VERTEX\_DATA : exact-integer?  
GLU\_TESS\_END\_DATA : exact-integer?  
GLU\_TESS\_ERROR\_DATA : exact-integer?  
GLU\_TESS\_EDGE\_FLAG\_DATA : exact-integer?  
GLU\_TESS\_COMBINE\_DATA : exact-integer?  
GLU\_CW : exact-integer?  
GLU\_CCW : exact-integer?  
GLU\_INTERIOR : exact-integer?  
GLU\_EXTERIOR : exact-integer?  
GLU\_UNKNOWN : exact-integer?  
GLU\_TESS\_WINDING\_RULE : exact-integer?  
GLU\_TESS\_BOUNDARY\_ONLY : exact-integer?  
GLU\_TESS\_TOLERANCE : exact-integer?  
GLU\_TESS\_ERROR1 : exact-integer?  
GLU\_TESS\_ERROR2 : exact-integer?  
GLU\_TESS\_ERROR3 : exact-integer?  
GLU\_TESS\_ERROR4 : exact-integer?  
GLU\_TESS\_ERROR5 : exact-integer?  
GLU\_TESS\_ERROR6 : exact-integer?  
GLU\_TESS\_ERROR7 : exact-integer?  
GLU\_TESS\_ERROR8 : exact-integer?

```
GLU_TESS_MISSING_BEGIN_POLYGON : exact-integer?  
GLU_TESS_MISSING_BEGIN_CONTOUR : exact-integer?  
GLU_TESS_MISSING_END_POLYGON : exact-integer?  
GLU_TESS_MISSING_END_CONTOUR : exact-integer?  
GLU_TESS_COORD_TOO_LARGE : exact-integer?  
GLU_TESS_NEED_COMBINE_CALLBACK : exact-integer?  
GLU_TESS_WINDING_ODD : exact-integer?  
GLU_TESS_WINDING_NONZERO : exact-integer?  
GLU_TESS_WINDING_POSITIVE : exact-integer?  
GLU_TESS_WINDING_NEGATIVE : exact-integer?  
GLU_TESS_WINDING_ABS_GEQ_TWO : exact-integer?  
GLU_TESS_MAX_COORD : real?
```

All OpenGL-defined constants.

```
(feedback-buffer->gl-float-vector buf) → gl-float-vector?  
  buf : feedback-buffer-object?
```

Converts a result from `glFeedbackBuffer` to a vector.

```
(select-buffer->gl-uint-vector buf) → gl-uint-vector?  
  buf : select-buffer-object?
```

Converts a result from `glSelectBuffer` to a vector.

### 3 Racket-Style OpenGL

```
(require sgl)      package: sgl
```

The functions in `sgl` use Racket style names instead of C style names. To convert a C OpenGL name to a Racket OpenGL name, change the `gl` prefix to `gl-`, separate adjacent words with hyphens, and convert to all lower case. Functions that have several variants to accommodate different numbers and types of arguments are collapsed into one or two functions in `sgl`. For example, `sgl` provides two vertex functions: `vertex` and `vertex-v`. The `vertex` function accepts 2, 3 or 4 numerical arguments, and the `vertex-v` function accepts `gl-vectors` of length 2, 3 or 4. The C language OpenGL interface, in contrast, has 24 vertex functions: `glVertex3i`, `glVertex4fv`, etc.

Functions in `sgl` take symbols instead of integers for `GLenum` arguments. Each function checks that the given symbol is an acceptable argument and raises an exception if it is not. Given the name of a C-language `#define` constant, determine the corresponding symbolic argument by removing the leading `GL_`, converting the letters to lower-case and replacing each `_` with `-`. For example, `GL_TRIANGLES` becomes `'triangles`, and `GL_TRIANGLE_STRIP` becomes `'triangle-strip`. Additionally, the functions check the length of any array arguments to ensure that OpenGL does not attempt to write or read after the array.

The `sgl` module is not as complete as the `sgl/gl` module.

Examples:

```
(require sgl sgl/gl-vectors)
(gl-begin 'triangles)
(gl-vertex 1 2 3)
(gl-vertex-v (gl-float-vector 1 2 3 4))
(gl-end)
```

```
(struct gl-selection-record (min-z max-z stack)
  #:extra-constructor-name make-gl-selection-record)
min-z : real?
max-z : real?
stack : ....
```

Represents a selection.

```
gl-accum : procedure?
gl-active-texture : procedure?
gl-alpha-func : procedure?
gl-begin : procedure?
gl-begin-query : procedure?
```



gl-blend-color : procedure?  
gl-blend-equation : procedure?  
gl-blend-func : procedure?  
gl-blend-func-separate : procedure?  
gl-call-list : procedure?  
gl-check-extension : procedure?  
gl-clear : procedure?  
gl-clear-accum : procedure?  
gl-clear-color : procedure?  
gl-clear-depth : procedure?  
gl-clear-index : procedure?  
gl-clear-stencil : procedure?  
gl-clip-plane : procedure?  
gl-color : procedure?  
gl-color-mask : procedure?  
gl-color-material : procedure?  
gl-color-v : procedure?  
gl-copy-pixels : procedure?  
gl-cull-face : procedure?  
gl-cylinder : procedure?  
gl-delete-lists : procedure?  
gl-delete-queries : procedure?  
gl-depth-func : procedure?  
gl-depth-mask : procedure?  
gl-depth-range : procedure?  
gl-disable : procedure?  
gl-disk : procedure?  
gl-edge-flag : procedure?  
gl-enable : procedure?  
gl-end : procedure?  
gl-end-list : procedure?  
gl-end-query : procedure?  
gl-eval-coord : procedure?  
gl-eval-coord-v : procedure?  
gl-eval-mesh : procedure?  
gl-eval-point : procedure?  
gl-feedback-buffer->gl-float-vector : procedure?  
gl-finish : procedure?  
gl-flush : procedure?  
gl-front-face : procedure?  
gl-frustum : procedure?  
gl-gen-lists : procedure?  
gl-gen-queries : procedure?  
gl-get-error : procedure?

gl-get-string : procedure?  
gl-hint : procedure?  
gl-index : procedure?  
gl-index-mask : procedure?  
gl-index-v : procedure?  
gl-init-names : procedure?  
gl-is-buffer : procedure?  
gl-is-enabled : procedure?  
gl-is-list : procedure?  
gl-is-query : procedure?  
gl-light : procedure?  
gl-light-model : procedure?  
gl-light-model-v : procedure?  
gl-light-v : procedure?  
gl-line-stipple : procedure?  
gl-line-width : procedure?  
gl-list-base : procedure?  
gl-load-identity : procedure?  
gl-load-matrix : procedure?  
gl-load-name : procedure?  
gl-load-transpose-matrix : procedure?  
gl-look-at : procedure?  
gl-map-grid : procedure?  
gl-material : procedure?  
gl-material-v : procedure?  
gl-matrix-mode : procedure?  
gl-mult-matrix : procedure?  
gl-mult-transpose-matrix : procedure?  
gl-multi-tex-coord : procedure?  
gl-multi-tex-coord-v : procedure?  
gl-new-list : procedure?  
gl-new-quadric : procedure?  
gl-normal : procedure?  
gl-normal-v : procedure?  
gl-ortho : procedure?  
gl-ortho-2d : procedure?  
gl-partial-disk : procedure?  
gl-pass-through : procedure?  
gl-perspective : procedure?  
gl-pick-matrix : procedure?  
gl-pixel-store : procedure?  
gl-point-parameter : procedure?  
gl-point-parameter-v : procedure?  
gl-point-size : procedure?

gl-polygon-mode : procedure?  
gl-polygon-offset : procedure?  
gl-pop-attrib : procedure?  
gl-pop-client-attrib : procedure?  
gl-pop-matrix : procedure?  
gl-pop-name : procedure?  
gl-project : procedure?  
gl-push-matrix : procedure?  
gl-push-name : procedure?  
gl-quadric-draw-style : procedure?  
gl-quadric-normals : procedure?  
gl-quadric-orientation : procedure?  
gl-quadric-texture : procedure?  
gl-raster-pos : procedure?  
gl-raster-pos-v : procedure?  
gl-rect : procedure?  
gl-rect-v : procedure?  
gl-render-mode : procedure?  
gl-rotate : procedure?  
gl-sample-coverage : procedure?  
gl-scale : procedure?  
gl-scissor : procedure?  
gl-secondary-color : procedure?  
gl-secondary-color-v : procedure?  
gl-select-buffer->gl-uint-vector : procedure?  
gl-shade-model : procedure?  
gl-sphere : procedure?  
gl-stencil-func : procedure?  
gl-stencil-mask : procedure?  
gl-stencil-op : procedure?  
gl-tex-coord : procedure?  
gl-tex-coord-v : procedure?  
gl-tex-gen : procedure?  
gl-tex-gen-v : procedure?  
gl-translate : procedure?  
gl-u-get-string : procedure?  
gl-un-project : procedure?  
gl-un-project4 : procedure?  
gl-vertex : procedure?  
gl-vertex-v : procedure?  
gl-viewport : procedure?  
gl-window-pos : procedure?  
gl-window-pos-v : procedure?

Racket-style variants of the OpenGL functions.

```
(gl-process-selection vec hits) → (listof gl-selection-record?)  
  vec : gl-uint-vector?  
  hits : exact-nonnegative-integer?
```

Parses the contents of *vec* from the format used by `glSelectBuffer`. The second argument should be the number of hits as returned by `glRenderMode`.

```
(gl-get-gl-version-number) → exact-nonnegative-integer?
```

Returns the run-time OpenGL version number as an integer: 10, 11, 12, 13, 14, 15, or 20.

```
(gl-get-glu-version-number) → exact-nonnegative-integer?
```

Returns the run-time GLU version number as an integer: 10, 11, 12, or 13.

## 4 OpenGL Vectors

```
(require sgl/gl-vectors)    package: sgl
```

The `sgl/gl-vectors` module supports OpenGL programming with `cvector`s. In this document and in the error messages, a “gl-vector” is just a `cvector`, while a “gl-*<type>*-vector” is a `cvector` with an appropriate type. Use the `sgl/gl-vectors` module vectors instead of a Racket `cvector` directly, because they are specialized to handle the OpenGL types correctly.

```
(gl-vector? v) → boolean?  
  v : any/c  
(gl-vector->vector vec) → vector?  
  vec : cvector?  
(gl-vector->list vec) → list?  
  vec : cvector?  
(gl-vector-length vec) → exact-nonnegative-integer?  
  vec : cvector?  
(gl-vector-ref vec pos) → any/v  
  vec : cvector?  
  pos : exact-nonnegative-integer?  
(gl-vector-set! vec pos v) → void?  
  vec : cvector?  
  pos : exact-nonnegative-integer?  
  v : any/v
```

Synonyms for `cvector?`, `cvector->vector`, `cvector-length`, etc.

```
(gl-byte-vector? v) → boolean?  
  v : any/c  
(make-gl-byte-vector pos) → gl-byte-vector?  
  pos : exact-nonnegative-integer?  
(gl-byte-vector v ...) → gl-byte-vector?  
  v : byte?  
(vector->gl-byte-vector v ...) → gl-byte-vector?  
  v : (vectorof byte?)  
(list->gl-byte-vector v ...) → gl-byte-vector?  
  v : (listof byte?)  
(gl-byte-vector+ vec ...+) → gl-byte-vector?  
  vec : gl-byte-vector?  
(gl-byte-vector- vec ...+) → gl-byte-vector?  
  vec : gl-byte-vector?  
(gl-byte-vector* x vec) → gl-byte-vector?  
  x : real?  
  vec : gl-byte-vector?
```

Operations on vectors of byte elements. The `gl-byte-vector+` and `gl-byte-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-byte-vector*` function multiplies each element of `vec` by `x`.

```
(gl-byte-vector? v) → boolean?
  v : any/c
(make-gl-byte-vector pos) → gl-byte-vector?
  pos : exact-nonnegative-integer?
(gl-byte-vector v ...) → gl-byte-vector?
  v : ubyte?
(vector->gl-byte-vector v ...) → gl-byte-vector?
  v : (vectorof ubyte?)
(list->gl-byte-vector v ...) → gl-byte-vector?
  v : (listof ubyte?)
(gl-byte-vector+ vec ...+) → gl-byte-vector?
  vec : gl-byte-vector?
(gl-byte-vector- vec ...+) → gl-byte-vector?
  vec : gl-byte-vector?
(gl-byte-vector* x vec) → gl-byte-vector?
  x : real?
  vec : gl-byte-vector?
```

Operations on vectors of ubyte elements. The `gl-ubyte-vector+` and `gl-ubyte-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-ubyte-vector*` function multiplies each element of `vec` by `x`.

```
(gl-short-vector? v) → boolean?
  v : any/c
(make-gl-short-vector pos) → gl-short-vector?
  pos : exact-nonnegative-integer?
(gl-short-vector v ...) → gl-short-vector?
  v : short?
(vector->gl-short-vector v ...) → gl-short-vector?
  v : (vectorof short?)
(list->gl-short-vector v ...) → gl-short-vector?
  v : (listof short?)
(gl-short-vector+ vec ...+) → gl-short-vector?
  vec : gl-short-vector?
(gl-short-vector- vec ...+) → gl-short-vector?
  vec : gl-short-vector?
(gl-short-vector* x vec) → gl-short-vector?
  x : real?
  vec : gl-short-vector?
```

Operations on vectors of short elements. The `gl-short-vector+` and `gl-short-`

`vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-short-vector*` function multiplies each element of `vec` by `x`.

```
(gl-ushort-vector? v) → boolean?  
  v : any/c  
(make-gl-ushort-vector pos) → gl-ushort-vector?  
  pos : exact-nonnegative-integer?  
(gl-ushort-vector v ...) → gl-ushort-vector?  
  v : ushort?  
(vector->gl-ushort-vector v ...) → gl-ushort-vector?  
  v : (vectorof ushort?)  
(list->gl-ushort-vector v ...) → gl-ushort-vector?  
  v : (listof ushort?)  
(gl-ushort-vector+ vec ...+) → gl-ushort-vector?  
  vec : gl-ushort-vector?  
(gl-ushort-vector- vec ...+) → gl-ushort-vector?  
  vec : gl-ushort-vector?  
(gl-ushort-vector* x vec) → gl-ushort-vector?  
  x : real?  
  vec : gl-ushort-vector?
```

Operations on vectors of `ushort` elements. The `gl-ushort-vector+` and `gl-ushort-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-ushort-vector*` function multiplies each element of `vec` by `x`.

```
(gl-int-vector? v) → boolean?  
  v : any/c  
(make-gl-int-vector pos) → gl-int-vector?  
  pos : exact-nonnegative-integer?  
(gl-int-vector v ...) → gl-int-vector?  
  v : int?  
(vector->gl-int-vector v ...) → gl-int-vector?  
  v : (vectorof int?)  
(list->gl-int-vector v ...) → gl-int-vector?  
  v : (listof int?)  
(gl-int-vector+ vec ...+) → gl-int-vector?  
  vec : gl-int-vector?  
(gl-int-vector- vec ...+) → gl-int-vector?  
  vec : gl-int-vector?  
(gl-int-vector* x vec) → gl-int-vector?  
  x : real?  
  vec : gl-int-vector?
```

Operations on vectors of `int` elements. The `gl-int-vector+` and `gl-int-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively.

The `gl-int-vector*` function multiplies each element of `vec` by `x`.

```
(gl-uint-vector? v) → boolean?  
  v : any/c  
(make-gl-uint-vector pos) → gl-uint-vector?  
  pos : exact-nonnegative-integer?  
(gl-uint-vector v ...) → gl-uint-vector?  
  v : uint?  
(vector->gl-uint-vector v ...) → gl-uint-vector?  
  v : (vectorof uint?)  
(list->gl-uint-vector v ...) → gl-uint-vector?  
  v : (listof uint?)  
(gl-uint-vector+ vec ...+) → gl-uint-vector?  
  vec : gl-uint-vector?  
(gl-uint-vector- vec ...+) → gl-uint-vector?  
  vec : gl-uint-vector?  
(gl-uint-vector* x vec) → gl-uint-vector?  
  x : real?  
  vec : gl-uint-vector?
```

Operations on vectors of `uint` elements. The `gl-uint-vector+` and `gl-uint-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-uint-vector*` function multiplies each element of `vec` by `x`.

```
(gl-float-vector? v) → boolean?  
  v : any/c  
(make-gl-float-vector pos) → gl-float-vector?  
  pos : exact-nonnegative-integer?  
(gl-float-vector v ...) → gl-float-vector?  
  v : float?  
(vector->gl-float-vector v ...) → gl-float-vector?  
  v : (vectorof float?)  
(list->gl-float-vector v ...) → gl-float-vector?  
  v : (listof float?)  
(gl-float-vector+ vec ...+) → gl-float-vector?  
  vec : gl-float-vector?  
(gl-float-vector- vec ...+) → gl-float-vector?  
  vec : gl-float-vector?  
(gl-float-vector* x vec) → gl-float-vector?  
  x : real?  
  vec : gl-float-vector?
```

Operations on vectors of `float` elements. The `gl-float-vector+` and `gl-float-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-float-vector*` function multiplies each element of `vec` by `x`.



```

(gl-double-vector? v) → boolean?
  v : any/c
(make-gl-double-vector pos) → gl-double-vector?
  pos : exact-nonnegative-integer?
(gl-double-vector v ...) → gl-double-vector?
  v : double?
(vector->gl-double-vector v ...) → gl-double-vector?
  v : (vectorof double?)
(list->gl-double-vector v ...) → gl-double-vector?
  v : (listof double?)
(gl-double-vector+ vec ...+) → gl-double-vector?
  vec : gl-double-vector?
(gl-double-vector- vec ...+) → gl-double-vector?
  vec : gl-double-vector?
(gl-double-vector* x vec) → gl-double-vector?
  x : real?
  vec : gl-double-vector?

```

Operations on vectors of double elements. The `gl-double-vector+` and `gl-double-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-double-vector*` function multiplies each element of `vec` by `x`.

```

(gl-boolean-vector? v) → boolean?
  v : any/c
(make-gl-boolean-vector pos) → gl-boolean-vector?
  pos : exact-nonnegative-integer?
(gl-boolean-vector v ...) → gl-boolean-vector?
  v : boolean?
(vector->gl-boolean-vector v ...) → gl-boolean-vector?
  v : (vectorof boolean?)
(list->gl-boolean-vector v ...) → gl-boolean-vector?
  v : (listof boolean?)
(gl-boolean-vector+ vec ...+) → gl-boolean-vector?
  vec : gl-boolean-vector?
(gl-boolean-vector- vec ...+) → gl-boolean-vector?
  vec : gl-boolean-vector?
(gl-boolean-vector* x vec) → gl-boolean-vector?
  x : real?
  vec : gl-boolean-vector?

```

Operations on vectors of boolean elements. The `gl-boolean-vector+` and `gl-boolean-vector-` functions compute the element-by-element sum and difference of the given vectors, respectively. The `gl-boolean-vector*` function multiplies each element of `vec` by `x`.

```
(gl-vector-norm vec) → real?  
vec : gl-vector?
```

Returns the square root of the sum of the squares of the elements of `vec`.

## 5 Bitmaps

```
(require sgl/bitmap)      package: sgl

(bitmap->gl-list bitmap
  [#:with-gl with-gl-proc
   #:mask mask])         → exact-integer?
bitmap : (is-a?/c bitmap%)
with-gl-proc : ((-> any) . -> . any) = (lambda (f) (f))
mask : (or/c (is-a?/c bitmap%) false/c)
      = (send bitmap get-loaded-mask)
```

Converts the given bitmap into an OpenGL list that can be rendered with `gl-call-list` or `glCallList`. The rendered object is a square on the  $z=0$  plane with corners at (0,0) and (1,1).

The `with-gl-proc` must accept a thunk and call it while the relevant OpenGL context is selected. Otherwise, the relevant OpenGL context must be selected already.

If `mask` is not `#f`, it is used as the mask bitmap for extracting alpha values.

## 6 Initialization

```
(require sgl/init)    package: sgl
```

Requiring the `sgl/init` library initializes platform-specific OpenGL state to help avoid crashes when OpenGL commands are incorrectly used without a current context. This library is required by `sgl` and `sgl/gl`, so it normally does not need to be required explicitly.

On Mac OS X, `sgl/init` checks whether any GL context is current, and if not, it creates a dummy context and sets it as the current context.

## Index

bitmap->gl-list, 51  
Bitmaps, 51  
C-Style OpenGL, 4  
feedback-buffer->gl-float-vector,  
39  
GFclampf, 4  
GFfloat, 4  
gl-accum, 40  
gl-active-texture, 40  
gl-alpha-func, 40  
gl-begin, 40  
gl-begin-query, 40  
gl-blend-color, 41  
gl-blend-equation, 41  
gl-blend-func, 41  
gl-blend-func-separate, 41  
gl-boolean-vector, 49  
gl-boolean-vector\*, 49  
gl-boolean-vector+, 49  
gl-boolean-vector-, 49  
gl-boolean-vector?, 49  
gl-byte-vector, 45  
gl-byte-vector\*, 45  
gl-byte-vector+, 45  
gl-byte-vector-, 45  
gl-byte-vector?, 45  
gl-call-list, 41  
gl-check-extension, 41  
gl-clear, 41  
gl-clear-accum, 41  
gl-clear-color, 41  
gl-clear-depth, 41  
gl-clear-index, 41  
gl-clear-stencil, 41  
gl-clip-plane, 41  
gl-color, 41  
gl-color-mask, 41  
gl-color-material, 41  
gl-color-v, 41  
gl-copy-pixels, 41  
gl-cull-face, 41  
gl-cylinder, 41  
gl-delete-lists, 41  
gl-delete-queries, 41  
gl-depth-func, 41  
gl-depth-mask, 41  
gl-depth-range, 41  
gl-disable, 41  
gl-disk, 41  
gl-double-vector, 49  
gl-double-vector\*, 49  
gl-double-vector+, 49  
gl-double-vector-, 49  
gl-double-vector?, 49  
gl-edge-flag, 41  
gl-enable, 41  
gl-end, 41  
gl-end-list, 41  
gl-end-query, 41  
gl-eval-coord, 41  
gl-eval-coord-v, 41  
gl-eval-mesh, 41  
gl-eval-point, 41  
gl-feedback-buffer->gl-float-  
vector, 41  
gl-finish, 41  
gl-float-vector, 48  
gl-float-vector\*, 48  
gl-float-vector+, 48  
gl-float-vector-, 48  
gl-float-vector?, 48  
gl-flush, 41  
gl-front-face, 41  
gl-frustum, 41  
gl-gen-lists, 41  
gl-gen-queries, 41  
gl-get-error, 41  
gl-get-gl-version-number, 44  
gl-get-glu-version-number, 44  
gl-get-string, 42  
gl-hint, 42  
gl-index, 42

[gl-index-mask](#), 42  
[gl-index-v](#), 42  
[gl-init-names](#), 42  
[gl-int-vector](#), 47  
[gl-int-vector\\*](#), 47  
[gl-int-vector+](#), 47  
[gl-int-vector-](#), 47  
[gl-int-vector?](#), 47  
[gl-is-buffer](#), 42  
[gl-is-enabled](#), 42  
[gl-is-list](#), 42  
[gl-is-query](#), 42  
[gl-light](#), 42  
[gl-light-model](#), 42  
[gl-light-model-v](#), 42  
[gl-light-v](#), 42  
[gl-line-stipple](#), 42  
[gl-line-width](#), 42  
[gl-list-base](#), 42  
[gl-load-identity](#), 42  
[gl-load-matrix](#), 42  
[gl-load-name](#), 42  
[gl-load-transpose-matrix](#), 42  
[gl-look-at](#), 42  
[gl-map-grid](#), 42  
[gl-material](#), 42  
[gl-material-v](#), 42  
[gl-matrix-mode](#), 42  
[gl-mult-matrix](#), 42  
[gl-mult-transpose-matrix](#), 42  
[gl-multi-tex-coord](#), 42  
[gl-multi-tex-coord-v](#), 42  
[gl-new-list](#), 42  
[gl-new-quadric](#), 42  
[gl-normal](#), 42  
[gl-normal-v](#), 42  
[gl-ortho](#), 42  
[gl-ortho-2d](#), 42  
[gl-partial-disk](#), 42  
[gl-pass-through](#), 42  
[gl-perspective](#), 42  
[gl-pick-matrix](#), 42  
[gl-pixel-store](#), 42  
[gl-point-parameter](#), 42  
[gl-point-parameter-v](#), 42  
[gl-point-size](#), 42  
[gl-polygon-mode](#), 43  
[gl-polygon-offset](#), 43  
[gl-pop-attrib](#), 43  
[gl-pop-client-attrib](#), 43  
[gl-pop-matrix](#), 43  
[gl-pop-name](#), 43  
[gl-process-selection](#), 44  
[gl-project](#), 43  
[gl-push-matrix](#), 43  
[gl-push-name](#), 43  
[gl-quadric-draw-style](#), 43  
[gl-quadric-normals](#), 43  
[gl-quadric-orientation](#), 43  
[gl-quadric-texture](#), 43  
[gl-raster-pos](#), 43  
[gl-raster-pos-v](#), 43  
[gl-rect](#), 43  
[gl-rect-v](#), 43  
[gl-render-mode](#), 43  
[gl-rotate](#), 43  
[gl-sample-coverage](#), 43  
[gl-scale](#), 43  
[gl-scissor](#), 43  
[gl-secondary-color](#), 43  
[gl-secondary-color-v](#), 43  
[gl-select-buffer->gl-uint-vector](#),  
43  
[gl-selection-record](#), 40  
[gl-selection-record-max-z](#), 40  
[gl-selection-record-min-z](#), 40  
[gl-selection-record-stack](#), 40  
[gl-selection-record?](#), 40  
[gl-shade-model](#), 43  
[gl-short-vector](#), 46  
[gl-short-vector\\*](#), 46  
[gl-short-vector+](#), 46  
[gl-short-vector-](#), 46  
[gl-short-vector?](#), 46

[gl-sphere](#), 43  
[gl-stencil-func](#), 43  
[gl-stencil-mask](#), 43  
[gl-stencil-op](#), 43  
[gl-tex-coord](#), 43  
[gl-tex-coord-v](#), 43  
[gl-tex-gen](#), 43  
[gl-tex-gen-v](#), 43  
[gl-translate](#), 43  
[gl-u-get-string](#), 43  
[gl-ubyte-vector](#), 46  
[gl-ubyte-vector\\*](#), 46  
[gl-ubyte-vector+](#), 46  
[gl-ubyte-vector-](#), 46  
[gl-ubyte-vector?](#), 46  
[gl-uint-vector](#), 48  
[gl-uint-vector\\*](#), 48  
[gl-uint-vector+](#), 48  
[gl-uint-vector-](#), 48  
[gl-uint-vector?](#), 48  
[gl-un-project](#), 43  
[gl-un-project4](#), 43  
[gl-ushort-vector](#), 47  
[gl-ushort-vector\\*](#), 47  
[gl-ushort-vector+](#), 47  
[gl-ushort-vector-](#), 47  
[gl-ushort-vector?](#), 47  
[gl-vector->list](#), 45  
[gl-vector->vector](#), 45  
[gl-vector-length](#), 45  
[gl-vector-norm](#), 50  
[gl-vector-ref](#), 45  
[gl-vector-set!](#), 45  
[gl-vector?](#), 45  
[gl-vertex](#), 43  
[gl-vertex-v](#), 43  
[gl-viewport](#), 43  
[gl-window-pos](#), 43  
[gl-window-pos-v](#), 43  
[GL: 3-D Graphics](#), 1  
[GL\\_2\\_BYTES](#), 17  
[GL\\_2D](#), 21  
[GL\\_3\\_BYTES](#), 17  
[GL\\_3D](#), 21  
[GL\\_3D\\_COLOR](#), 21  
[GL\\_3D\\_COLOR\\_TEXTURE](#), 21  
[GL\\_4\\_BYTES](#), 17  
[GL\\_4D\\_COLOR\\_TEXTURE](#), 21  
[GL\\_ACCUM](#), 20  
[GL\\_ACCUM\\_ALPHA\\_BITS](#), 20  
[GL\\_ACCUM\\_BLUE\\_BITS](#), 20  
[GL\\_ACCUM\\_BUFFER\\_BIT](#), 27  
[GL\\_ACCUM\\_CLEAR\\_VALUE](#), 20  
[GL\\_ACCUM\\_GREEN\\_BITS](#), 20  
[GL\\_ACCUM\\_RED\\_BITS](#), 20  
[GL\\_ACTIVE\\_TEXTURE](#), 32  
[GL\\_ADD](#), 20  
[GL\\_ADD\\_SIGNED](#), 33  
[GL\\_ALIASED\\_LINE\\_WIDTH\\_RANGE](#), 29  
[GL\\_ALIASED\\_POINT\\_SIZE\\_RANGE](#), 29  
[GL\\_ALL\\_ATTRIB\\_BITS](#), 28  
[GL\\_ALL\\_CLIENT\\_ATTRIB\\_BITS](#), 29  
[GL\\_ALPHA](#), 23  
[GL\\_ALPHA12](#), 28  
[GL\\_ALPHA16](#), 28  
[GL\\_ALPHA4](#), 28  
[GL\\_ALPHA8](#), 28  
[GL\\_ALPHA\\_BIAS](#), 25  
[GL\\_ALPHA\\_BITS](#), 23  
[GL\\_ALPHA\\_SCALE](#), 25  
[GL\\_ALPHA\\_TEST](#), 20  
[GL\\_ALPHA\\_TEST\\_FUNC](#), 20  
[GL\\_ALPHA\\_TEST\\_REF](#), 20  
[GL\\_ALWAYS](#), 19  
[GL\\_AMBIENT](#), 20  
[GL\\_AMBIENT\\_AND\\_DIFFUSE](#), 20  
[GL\\_AND](#), 22  
[GL\\_AND\\_INVERTED](#), 22  
[GL\\_AND\\_REVERSE](#), 22  
[GL\\_ARRAY\\_BUFFER](#), 34  
[GL\\_ARRAY\\_BUFFER\\_BINDING](#), 34  
[GL\\_ATTRIB\\_STACK\\_DEPTH](#), 23  
[GL\\_AUTO\\_NORMAL](#), 24  
[GL\\_AUX0](#), 22

GL\_AUX1, 22  
 GL\_AUX2, 22  
 GL\_AUX3, 22  
 GL\_AUX\_BUFFERS, 23  
 GL\_BACK, 18  
 GL\_BACK\_LEFT, 22  
 GL\_BACK\_RIGHT, 22  
 GL\_BGR, 29  
 GL\_BGRA, 29  
 GL\_BITMAP, 23  
 GL\_BITMAP\_TOKEN, 21  
 GL\_BLEND, 20  
 GL\_BLEND\_COLOR, 30  
 GL\_BLEND\_DST, 20  
 GL\_BLEND\_DST\_ALPHA, 33  
 GL\_BLEND\_DST\_RGB, 33  
 GL\_BLEND\_EQUATION, 30  
 GL\_BLEND\_SRC, 20  
 GL\_BLEND\_SRC\_ALPHA, 33  
 GL\_BLEND\_SRC\_RGB, 33  
 GL\_BLUE, 23  
 GL\_BLUE\_BIAS, 25  
 GL\_BLUE\_BITS, 23  
 GL\_BLUE\_SCALE, 25  
 GL\_BUFFER\_ACCESS, 35  
 GL\_BUFFER\_MAP\_POINTER, 35  
 GL\_BUFFER\_MAPPED, 35  
 GL\_BUFFER\_SIZE, 34  
 GL\_BUFFER\_USAGE, 34  
 GL\_BYTE, 17  
 GL\_C3F\_V3F, 18  
 GL\_C4F\_N3F\_V3F, 18  
 GL\_C4UB\_V2F, 18  
 GL\_C4UB\_V3F, 18  
 GL\_CCW, 18  
 GL\_CLAMP, 27  
 GL\_CLAMP\_TO\_BORDER, 33  
 GL\_CLAMP\_TO\_EDGE, 29  
 GL\_CLEAR, 22  
 GL\_CLIENT\_ACTIVE\_TEXTURE, 32  
 GL\_CLIENT\_ALL\_ATTRIB\_BITS, 29  
 GL\_CLIENT\_ATTRIB\_STACK\_DEPTH, 23  
 GL\_CLIENT\_PIXEL\_STORE\_BIT, 28  
 GL\_CLIENT\_VERTEX\_ARRAY\_BIT, 28  
 GL\_CLIP\_PLANE0, 20  
 GL\_CLIP\_PLANE1, 20  
 GL\_CLIP\_PLANE2, 20  
 GL\_CLIP\_PLANE3, 20  
 GL\_CLIP\_PLANE4, 20  
 GL\_CLIP\_PLANE5, 20  
 GL\_COEFF, 24  
 GL\_COLOR, 23  
 GL\_COLOR\_ARRAY, 17  
 GL\_COLOR\_ARRAY\_BUFFER\_BINDING, 35  
 GL\_COLOR\_ARRAY\_POINTER, 18  
 GL\_COLOR\_ARRAY\_SIZE, 17  
 GL\_COLOR\_ARRAY\_STRIDE, 17  
 GL\_COLOR\_ARRAY\_TYPE, 17  
 GL\_COLOR\_BUFFER\_BIT, 27  
 GL\_COLOR\_CLEAR\_VALUE, 23  
 GL\_COLOR\_INDEX, 22  
 GL\_COLOR\_INDEXES, 20  
 GL\_COLOR\_LOGIC\_OP, 21  
 GL\_COLOR\_MATERIAL, 20  
 GL\_COLOR\_MATERIAL\_FACE, 20  
 GL\_COLOR\_MATERIAL\_PARAMETER, 20  
 GL\_COLOR\_MATRIX, 31  
 GL\_COLOR\_MATRIX\_STACK\_DEPTH, 31  
 GL\_COLOR\_SUM, 34  
 GL\_COLOR\_TABLE, 31  
 GL\_COLOR\_TABLE\_ALPHA\_SIZE, 31  
 GL\_COLOR\_TABLE\_BIAS, 31  
 GL\_COLOR\_TABLE\_BLUE\_SIZE, 31  
 GL\_COLOR\_TABLE\_FORMAT, 31  
 GL\_COLOR\_TABLE\_GREEN\_SIZE, 31  
 GL\_COLOR\_TABLE\_INTENSITY\_SIZE, 31  
 GL\_COLOR\_TABLE\_LUMINANCE\_SIZE, 31  
 GL\_COLOR\_TABLE\_RED\_SIZE, 31  
 GL\_COLOR\_TABLE\_SCALE, 31  
 GL\_COLOR\_TABLE\_WIDTH, 31  
 GL\_COLOR\_WRITEMASK, 23  
 GL\_COMBINE, 33  
 GL\_COMBINE\_ALPHA, 33  
 GL\_COMBINE\_RGB, 33



GL_COMPARE_R_TO_TEXTURE, 34	GL_CURRENT_RASTER_TEXTURE_COORDS,
GL_COMPILE, 19	24
GL_COMPILE_AND_EXECUTE, 19	GL_CURRENT_SECONDARY_COLOR, 34
GL_COMPRESSED_ALPHA, 33	GL_CURRENT_TEXTURE_COORDS, 24
GL_COMPRESSED_INTENSITY, 33	GL_CW, 18
GL_COMPRESSED_LUMINANCE, 33	GL_DECAL, 27
GL_COMPRESSED_LUMINANCE_ALPHA, 33	GL_DECR, 22
GL_COMPRESSED_RGB, 33	GL_DECR_WRAP, 34
GL_COMPRESSED_RGBA, 33	GL_DEPTH, 23
GL_COMPRESSED_TEXTURE_FORMATS, 33	GL_DEPTH_BIAS, 25
GL_CONSTANT, 33	GL_DEPTH_BITS, 19
GL_CONSTANT_ALPHA, 30	GL_DEPTH_BUFFER_BIT, 27
GL_CONSTANT_ATTENUATION, 19	GL_DEPTH_CLEAR_VALUE, 19
GL_CONSTANT_BORDER, 31	GL_DEPTH_COMPONENT, 19
GL_CONSTANT_COLOR, 29	GL_DEPTH_COMPONENT16, 34
GL_CONVOLUTION_1D, 30	GL_DEPTH_COMPONENT24, 34
GL_CONVOLUTION_2D, 30	GL_DEPTH_COMPONENT32, 34
GL_CONVOLUTION_BORDER_COLOR, 31	GL_DEPTH_FUNC, 19
GL_CONVOLUTION_BORDER_MODE, 30	GL_DEPTH_RANGE, 19
GL_CONVOLUTION_FILTER_BIAS, 30	GL_DEPTH_SCALE, 25
GL_CONVOLUTION_FILTER_SCALE, 30	GL_DEPTH_TEST, 19
GL_CONVOLUTION_FORMAT, 30	GL_DEPTH_TEXTURE_MODE, 34
GL_CONVOLUTION_HEIGHT, 30	GL_DEPTH_WRITEMASK, 19
GL_CONVOLUTION_WIDTH, 30	GL_DIFFUSE, 20
GL_COPY, 22	GL_DITHER, 23
GL_COPY_INVERTED, 22	GL_DOMAIN, 24
GL_COPY_PIXEL_TOKEN, 21	GL_DONT_CARE, 25
GL_CULL_FACE, 19	GL_DOT3_RGB, 33
GL_CULL_FACE_MODE, 19	GL_DOT3_RGBA, 33
GL_CURRENT_BIT, 27	GL_DOUBLE, 17
GL_CURRENT_COLOR, 23	GL_DOUBLEBUFFER, 23
GL_CURRENT_FOG_COORD, 35	GL_DRAW_BUFFER, 23
GL_CURRENT_FOG_COORDINATE, 34	GL_DRAW_PIXEL_TOKEN, 21
GL_CURRENT_INDEX, 23	GL_DST_ALPHA, 21
GL_CURRENT_NORMAL, 23	GL_DST_COLOR, 21
GL_CURRENT_QUERY, 34	GL_DYNAMIC_COPY, 35
GL_CURRENT_RASTER_COLOR, 23	GL_DYNAMIC_DRAW, 35
GL_CURRENT_RASTER_DISTANCE, 24	GL_DYNAMIC_READ, 35
GL_CURRENT_RASTER_INDEX, 24	GL_EDGE_FLAG, 19
GL_CURRENT_RASTER_POSITION, 24	GL_EDGE_FLAG_ARRAY, 17
GL_CURRENT_RASTER_POSITION_VALID,	GL_EDGE_FLAG_ARRAY_BUFFER_BINDING,
24	35

GL_EDGE_FLAG_ARRAY_POINTER, 18	GL_FOG_COORDINATE_SOURCE, 34
GL_EDGE_FLAG_ARRAY_STRIDE, 17	GL_FOG_DENSITY, 21
GL_ELEMENT_ARRAY_BUFFER, 34	GL_FOG_END, 21
GL_ELEMENT_ARRAY_BUFFER_BINDING, 34	GL_FOG_HINT, 25
GL_EMISSION, 20	GL_FOG_INDEX, 21
GL_ENABLE_BIT, 27	GL_FOG_MODE, 21
GL_EQUAL, 19	GL_FOG_START, 21
GL_EQUIV, 22	GL_FRAGMENT_DEPTH, 34
GL_EVAL_BIT, 27	GL_FRONT, 18
GL_EXP, 21	GL_FRONT_AND_BACK, 20
GL_EXP2, 21	GL_FRONT_FACE, 19
GL_EXTENSIONS, 27	GL_FRONT_LEFT, 22
GL_EYE_LINEAR, 27	GL_FRONT_RIGHT, 22
GL_EYE_PLANE, 27	GL_FUNC_ADD, 30
GL_FALSE, 17	GL_FUNC_REVERSE_SUBTRACT, 30
GL_FALSE, 4	GL_FUNC_SUBTRACT, 30
GL_FASTEST, 25	GL_GENERATE_MIPMAP, 34
GL_FEEDBACK, 21	GL_GENERATE_MIPMAP_HINT, 34
GL_FEEDBACK_BUFFER_POINTER, 21	GL_GEQUAL, 19
GL_FEEDBACK_BUFFER_SIZE, 21	GL_GREATER, 19
GL_FEEDBACK_BUFFER_TYPE, 21	GL_GREEN, 23
GL_FILL, 18	GL_GREEN_BIAS, 25
GL_FLAT, 20	GL_GREEN_BITS, 23
GL_FLOAT, 17	GL_GREEN_SCALE, 25
GL_FOG, 21	GL_HINT_BIT, 27
GL_FOG_BIT, 27	GL_HISTOGRAM, 30
GL_FOG_COLOR, 21	GL_HISTOGRAM_ALPHA_SIZE, 30
GL_FOG_COORD, 35	GL_HISTOGRAM_BLUE_SIZE, 30
GL_FOG_COORD_ARRAY, 35	GL_HISTOGRAM_FORMAT, 30
GL_FOG_COORD_ARRAY_BUFFER_BINDING, 35	GL_HISTOGRAM_GREEN_SIZE, 30
GL_FOG_COORD_ARRAY_POINTER, 35	GL_HISTOGRAM_LUMINANCE_SIZE, 30
GL_FOG_COORD_ARRAY_STRIDE, 35	GL_HISTOGRAM_RED_SIZE, 30
GL_FOG_COORD_ARRAY_TYPE, 35	GL_HISTOGRAM_SINK, 30
GL_FOG_COORD_SRC, 35	GL_HISTOGRAM_WIDTH, 30
GL_FOG_COORDINATE, 34	GL_INCR, 22
GL_FOG_COORDINATE_ARRAY, 34	GL_INCR_WRAP, 34
GL_FOG_COORDINATE_ARRAY_BUFFER_BINDING, 35	GL_INDEX_ARRAY, 17
GL_FOG_COORDINATE_ARRAY_POINTER, 34	GL_INDEX_ARRAY_BUFFER_BINDING, 35
GL_FOG_COORDINATE_ARRAY_STRIDE, 34	GL_INDEX_ARRAY_POINTER, 18
GL_FOG_COORDINATE_ARRAY_TYPE, 34	GL_INDEX_ARRAY_STRIDE, 17
	GL_INDEX_ARRAY_TYPE, 17
	GL_INDEX_BITS, 23

GL\_INDEX\_CLEAR\_VALUE, 24  
 GL\_INDEX\_LOGIC\_OP, 21  
 GL\_INDEX\_MODE, 24  
 GL\_INDEX\_OFFSET, 25  
 GL\_INDEX\_SHIFT, 25  
 GL\_INDEX\_WRITEMASK, 24  
 GL\_INT, 17  
 GL\_INTENSITY, 28  
 GL\_INTENSITY12, 28  
 GL\_INTENSITY16, 28  
 GL\_INTENSITY4, 28  
 GL\_INTENSITY8, 28  
 GL\_INTERPOLATE, 33  
 GL\_INVALID\_ENUM, 27  
 GL\_INVALID\_OPERATION, 27  
 GL\_INVALID\_VALUE, 27  
 GL\_INVERT, 22  
 GL\_KEEP, 22  
 GL\_LEFT, 22  
 GL\_LEQUAL, 19  
 GL\_LESS, 19  
 GL\_LIGHT0, 19  
 GL\_LIGHT1, 19  
 GL\_LIGHT2, 19  
 GL\_LIGHT3, 19  
 GL\_LIGHT4, 19  
 GL\_LIGHT5, 19  
 GL\_LIGHT6, 19  
 GL\_LIGHT7, 19  
 GL\_LIGHT\_MODEL\_AMBIENT, 20  
 GL\_LIGHT\_MODEL\_COLOR\_CONTROL, 29  
 GL\_LIGHT\_MODEL\_LOCAL\_VIEWER, 20  
 GL\_LIGHT\_MODEL\_TWO\_SIDE, 20  
 GL\_LIGHTING, 19  
 GL\_LIGHTING\_BIT, 27  
 GL\_LINE, 18  
 GL\_LINE\_BIT, 27  
 GL\_LINE\_LOOP, 17  
 GL\_LINE\_RESET\_TOKEN, 21  
 GL\_LINE\_SMOOTH, 18  
 GL\_LINE\_SMOOTH\_HINT, 25  
 GL\_LINE\_STIPPLE, 18  
 GL\_LINE\_STIPPLE\_PATTERN, 18  
 GL\_LINE\_STIPPLE\_REPEAT, 18  
 GL\_LINE\_STRIP, 17  
 GL\_LINE\_TOKEN, 21  
 GL\_LINE\_WIDTH, 18  
 GL\_LINE\_WIDTH\_GRANULARITY, 18  
 GL\_LINE\_WIDTH\_RANGE, 18  
 GL\_LINEAR, 21  
 GL\_LINEAR\_ATTENUATION, 19  
 GL\_LINEAR\_MIPMAP\_LINEAR, 26  
 GL\_LINEAR\_MIPMAP\_NEAREST, 26  
 GL\_LINES, 17  
 GL\_LIST\_BASE, 19  
 GL\_LIST\_BIT, 27  
 GL\_LIST\_INDEX, 19  
 GL\_LIST\_MODE, 19  
 GL\_LOAD, 20  
 GL\_LOGIC\_OP, 21  
 GL\_LOGIC\_OP\_MODE, 21  
 GL\_LUMINANCE, 23  
 GL\_LUMINANCE12, 28  
 GL\_LUMINANCE12\_ALPHA12, 28  
 GL\_LUMINANCE12\_ALPHA4, 28  
 GL\_LUMINANCE16, 28  
 GL\_LUMINANCE16\_ALPHA16, 28  
 GL\_LUMINANCE4, 28  
 GL\_LUMINANCE4\_ALPHA4, 28  
 GL\_LUMINANCE6\_ALPHA2, 28  
 GL\_LUMINANCE8, 28  
 GL\_LUMINANCE8\_ALPHA8, 28  
 GL\_LUMINANCE\_ALPHA, 23  
 GL\_MAP1\_COLOR\_4, 24  
 GL\_MAP1\_GRID\_DOMAIN, 24  
 GL\_MAP1\_GRID\_SEGMENTS, 24  
 GL\_MAP1\_INDEX, 24  
 GL\_MAP1\_NORMAL, 24  
 GL\_MAP1\_TEXTURE\_COORD\_1, 24  
 GL\_MAP1\_TEXTURE\_COORD\_2, 24  
 GL\_MAP1\_TEXTURE\_COORD\_3, 24  
 GL\_MAP1\_TEXTURE\_COORD\_4, 24  
 GL\_MAP1\_VERTEX\_3, 24  
 GL\_MAP1\_VERTEX\_4, 24

GL\_MAP2\_COLOR\_4, 24  
 GL\_MAP2\_GRID\_DOMAIN, 24  
 GL\_MAP2\_GRID\_SEGMENTS, 24  
 GL\_MAP2\_INDEX, 24  
 GL\_MAP2\_NORMAL, 24  
 GL\_MAP2\_TEXTURE\_COORD\_1, 24  
 GL\_MAP2\_TEXTURE\_COORD\_2, 24  
 GL\_MAP2\_TEXTURE\_COORD\_3, 24  
 GL\_MAP2\_TEXTURE\_COORD\_4, 24  
 GL\_MAP2\_VERTEX\_3, 24  
 GL\_MAP2\_VERTEX\_4, 24  
 GL\_MAP\_COLOR, 25  
 GL\_MAP\_STENCIL, 25  
 GL\_MATRIX\_MODE, 18  
 GL\_MAX, 30  
 GL\_MAX\_3D\_TEXTURE\_SIZE, 29  
 GL\_MAX\_ATTRIB\_STACK\_DEPTH, 23  
 GL\_MAX\_CLIENT\_ATTRIB\_STACK\_DEPTH, 23  
 GL\_MAX\_CLIP\_PLANES, 23  
 GL\_MAX\_COLOR\_MATRIX\_STACK\_DEPTH, 31  
 GL\_MAX\_CONVOLUTION\_HEIGHT, 30  
 GL\_MAX\_CONVOLUTION\_WIDTH, 30  
 GL\_MAX\_CUBE\_MAP\_TEXTURE\_SIZE, 33  
 GL\_MAX\_ELEMENTS\_INDICES, 29  
 GL\_MAX\_ELEMENTS\_VERTICES, 29  
 GL\_MAX\_EVAL\_ORDER, 23  
 GL\_MAX\_LIGHTS, 23  
 GL\_MAX\_LIST\_NESTING, 23  
 GL\_MAX\_MODELVIEW\_STACK\_DEPTH, 23  
 GL\_MAX\_NAME\_STACK\_DEPTH, 23  
 GL\_MAX\_PIXEL\_MAP\_TABLE, 23  
 GL\_MAX\_PROJECTION\_STACK\_DEPTH, 23  
 GL\_MAX\_TEXTURE\_LOD\_BIAS, 34  
 GL\_MAX\_TEXTURE\_SIZE, 23  
 GL\_MAX\_TEXTURE\_STACK\_DEPTH, 23  
 GL\_MAX\_TEXTURE\_UNITS, 32  
 GL\_MAX\_VIEWPORT\_DIMS, 23  
 GL\_MIN, 30  
 GL\_MINMAX, 30  
 GL\_MINMAX\_FORMAT, 30  
 GL\_MINMAX\_SINK, 30  
 GL\_MIRRORED\_REPEAT, 34  
 GL\_MODELVIEW, 18  
 GL\_MODELVIEW\_MATRIX, 24  
 GL\_MODELVIEW\_STACK\_DEPTH, 24  
 GL\_MODULATE, 27  
 GL\_MULT, 20  
 GL\_MULTISAMPLE, 32  
 GL\_MULTISAMPLE\_BIT, 32  
 GL\_N3F\_V3F, 18  
 GL\_NAME\_STACK\_DEPTH, 24  
 GL\_NAND, 22  
 GL\_NEAREST, 27  
 GL\_NEAREST\_MIPMAP\_LINEAR, 26  
 GL\_NEAREST\_MIPMAP\_NEAREST, 26  
 GL\_NEVER, 19  
 GL\_NICEST, 25  
 GL\_NO\_ERROR, 27  
 GL\_NONE, 22  
 GL\_NOOP, 22  
 GL\_NOR, 22  
 GL\_NORMAL\_ARRAY, 17  
 GL\_NORMAL\_ARRAY\_BUFFER\_BINDING, 35  
 GL\_NORMAL\_ARRAY\_POINTER, 18  
 GL\_NORMAL\_ARRAY\_STRIDE, 17  
 GL\_NORMAL\_ARRAY\_TYPE, 17  
 GL\_NORMAL\_MAP, 32  
 GL\_NORMALIZE, 20  
 GL\_NOTEQUAL, 19  
 GL\_NUM\_COMPRESSED\_TEXTURE\_FORMATS, 33  
 GL\_OBJECT\_LINEAR, 26  
 GL\_OBJECT\_PLANE, 26  
 GL\_ONE, 20  
 GL\_ONE\_MINUS\_CONSTANT\_ALPHA, 30  
 GL\_ONE\_MINUS\_CONSTANT\_COLOR, 30  
 GL\_ONE\_MINUS\_DST\_ALPHA, 21  
 GL\_ONE\_MINUS\_DST\_COLOR, 21  
 GL\_ONE\_MINUS\_SRC\_ALPHA, 21  
 GL\_ONE\_MINUS\_SRC\_COLOR, 21  
 GL\_OPERAND0\_ALPHA, 33  
 GL\_OPERAND0\_RGB, 33  
 GL\_OPERAND1\_ALPHA, 33

GL\_OPERAND1\_RGB, 33  
 GL\_OPERAND2\_ALPHA, 33  
 GL\_OPERAND2\_RGB, 33  
 GL\_OR, 22  
 GL\_OR\_INVERTED, 22  
 GL\_OR\_REVERSE, 22  
 GL\_ORDER, 25  
 GL\_OUT\_OF\_MEMORY, 27  
 GL\_PACK\_ALIGNMENT, 26  
 GL\_PACK\_IMAGE\_HEIGHT, 29  
 GL\_PACK\_LSB\_FIRST, 26  
 GL\_PACK\_ROW\_LENGTH, 26  
 GL\_PACK\_SKIP\_IMAGES, 29  
 GL\_PACK\_SKIP\_PIXELS, 26  
 GL\_PACK\_SKIP\_ROWS, 26  
 GL\_PACK\_SWAP\_BYTES, 26  
 GL\_PASS\_THROUGH\_TOKEN, 21  
 GL\_PERSPECTIVE\_CORRECTION\_HINT, 25  
 GL\_PIXEL\_MAP\_A\_TO\_A, 26  
 GL\_PIXEL\_MAP\_A\_TO\_A\_SIZE, 25  
 GL\_PIXEL\_MAP\_B\_TO\_B, 25  
 GL\_PIXEL\_MAP\_B\_TO\_B\_SIZE, 25  
 GL\_PIXEL\_MAP\_G\_TO\_G, 25  
 GL\_PIXEL\_MAP\_G\_TO\_G\_SIZE, 25  
 GL\_PIXEL\_MAP\_I\_TO\_A, 25  
 GL\_PIXEL\_MAP\_I\_TO\_A\_SIZE, 25  
 GL\_PIXEL\_MAP\_I\_TO\_B, 25  
 GL\_PIXEL\_MAP\_I\_TO\_B\_SIZE, 25  
 GL\_PIXEL\_MAP\_I\_TO\_G, 25  
 GL\_PIXEL\_MAP\_I\_TO\_G\_SIZE, 25  
 GL\_PIXEL\_MAP\_I\_TO\_I, 25  
 GL\_PIXEL\_MAP\_I\_TO\_I\_SIZE, 25  
 GL\_PIXEL\_MAP\_I\_TO\_R, 25  
 GL\_PIXEL\_MAP\_I\_TO\_R\_SIZE, 25  
 GL\_PIXEL\_MAP\_R\_TO\_R, 25  
 GL\_PIXEL\_MAP\_R\_TO\_R\_SIZE, 25  
 GL\_PIXEL\_MAP\_S\_TO\_S, 25  
 GL\_PIXEL\_MAP\_S\_TO\_S\_SIZE, 25  
 GL\_PIXEL\_MODE\_BIT, 27  
 GL\_POINT, 18  
 GL\_POINT\_BIT, 27  
 GL\_POINT\_DISTANCE\_ATTENUATION, 34  
 GL\_POINT\_FADE\_THRESHOLD\_SIZE, 34  
 GL\_POINT\_SIZE, 18  
 GL\_POINT\_SIZE\_GRANULARITY, 18  
 GL\_POINT\_SIZE\_MAX, 33  
 GL\_POINT\_SIZE\_MIN, 33  
 GL\_POINT\_SIZE\_RANGE, 18  
 GL\_POINT\_SMOOTH, 18  
 GL\_POINT\_SMOOTH\_HINT, 25  
 GL\_POINT\_TOKEN, 21  
 GL\_POINTS, 17  
 GL\_POLYGON, 17  
 GL\_POLYGON\_BIT, 27  
 GL\_POLYGON\_MODE, 18  
 GL\_POLYGON\_OFFSET\_FACTOR, 19  
 GL\_POLYGON\_OFFSET\_FILL, 19  
 GL\_POLYGON\_OFFSET\_LINE, 19  
 GL\_POLYGON\_OFFSET\_POINT, 19  
 GL\_POLYGON\_OFFSET\_UNITS, 19  
 GL\_POLYGON\_SMOOTH, 18  
 GL\_POLYGON\_SMOOTH\_HINT, 25  
 GL\_POLYGON\_STIPPLE, 19  
 GL\_POLYGON\_STIPPLE\_BIT, 27  
 GL\_POLYGON\_TOKEN, 21  
 GL\_POSITION, 20  
 GL\_POST\_COLOR\_MATRIX\_ALPHA\_BIAS, 31  
 GL\_POST\_COLOR\_MATRIX\_ALPHA\_SCALE, 31  
 GL\_POST\_COLOR\_MATRIX\_BLUE\_BIAS, 31  
 GL\_POST\_COLOR\_MATRIX\_BLUE\_SCALE, 31  
 GL\_POST\_COLOR\_MATRIX\_COLOR\_TABLE, 31  
 GL\_POST\_COLOR\_MATRIX\_GREEN\_BIAS, 31  
 GL\_POST\_COLOR\_MATRIX\_GREEN\_SCALE, 31  
 GL\_POST\_COLOR\_MATRIX\_RED\_BIAS, 31  
 GL\_POST\_COLOR\_MATRIX\_RED\_SCALE, 31  
 GL\_POST\_CONVOLUTION\_ALPHA\_BIAS, 30  
 GL\_POST\_CONVOLUTION\_ALPHA\_SCALE, 30  
 GL\_POST\_CONVOLUTION\_BLUE\_BIAS, 30  
 GL\_POST\_CONVOLUTION\_BLUE\_SCALE, 30  
 GL\_POST\_CONVOLUTION\_COLOR\_TABLE, 31  
 GL\_POST\_CONVOLUTION\_GREEN\_BIAS, 30

GL\_POST\_CONVOLUTION\_GREEN\_SCALE, 30  
 GL\_POST\_CONVOLUTION\_RED\_BIAS, 30  
 GL\_POST\_CONVOLUTION\_RED\_SCALE, 30  
 GL\_PREVIOUS, 33  
 GL\_PRIMARY\_COLOR, 33  
 GL\_PROJECTION, 18  
 GL\_PROJECTION\_MATRIX, 24  
 GL\_PROJECTION\_STACK\_DEPTH, 24  
 GL\_PROXY\_COLOR\_TABLE, 31  
 GL\_PROXY\_HISTOGRAM, 30  
 GL\_PROXY\_POST\_COLOR\_MATRIX\_COLOR\_TABLE, 31  
 GL\_PROXY\_POST\_CONVOLUTION\_COLOR\_TABLE, 31  
 GL\_PROXY\_TEXTURE\_1D, 28  
 GL\_PROXY\_TEXTURE\_2D, 28  
 GL\_PROXY\_TEXTURE\_3D, 29  
 GL\_PROXY\_TEXTURE\_CUBE\_MAP, 33  
 GL\_Q, 27  
 GL\_QUAD\_STRIP, 17  
 GL\_QUADRATIC\_ATTENUATION, 19  
 GL\_QUADS, 17  
 GL\_QUERY\_COUNTER\_BITS, 34  
 GL\_QUERY\_RESULT, 34  
 GL\_QUERY\_RESULT\_AVAILABLE, 34  
 GL\_R, 27  
 GL\_R3\_G3\_B2, 28  
 GL\_READ\_BUFFER, 23  
 GL\_READ\_ONLY, 35  
 GL\_READ\_WRITE, 35  
 GL\_RED, 22  
 GL\_RED\_BIAS, 25  
 GL\_RED\_BITS, 23  
 GL\_RED\_SCALE, 25  
 GL\_REDUCE, 30  
 GL\_REFLECTION\_MAP, 32  
 GL\_RENDER, 21  
 GL\_RENDER\_MODE, 24  
 GL\_RENDERER, 27  
 GL\_REPEAT, 27  
 GL\_REPLACE, 22  
 GL\_REPLICATE\_BORDER, 31  
 GL\_RESCALE\_NORMAL, 29  
 GL\_RETURN, 20  
 GL\_RGB, 23  
 GL\_RGB10, 28  
 GL\_RGB10\_A2, 28  
 GL\_RGB12, 28  
 GL\_RGB16, 28  
 GL\_RGB4, 28  
 GL\_RGB5, 28  
 GL\_RGB5\_A1, 28  
 GL\_RGB5\_A2, 28  
 GL\_RGB8, 28  
 GL\_RGBA, 23  
 GL\_RGBA12, 28  
 GL\_RGBA16, 28  
 GL\_RGBA2, 28  
 GL\_RGBA4, 28  
 GL\_RGBA8, 28  
 GL\_RGBA\_MODE, 24  
 GL\_RIGHT, 22  
 GL\_S, 27  
 GL\_SAMPLE\_ALPHA\_TO\_COVERAGE, 32  
 GL\_SAMPLE\_ALPHA\_TO\_ONE, 32  
 GL\_SAMPLE\_BUFFERS, 32  
 GL\_SAMPLE\_COVERAGE, 32  
 GL\_SAMPLE\_COVERAGE\_INVERT, 32  
 GL\_SAMPLE\_COVERAGE\_VALUE, 32  
 GL\_SAMPLES, 32  
 GL\_SAMPLES\_PASSED, 35  
 GL\_SCISSOR\_BIT, 28  
 GL\_SCISSOR\_BOX, 25  
 GL\_SCISSOR\_TEST, 25  
 GL\_SECONDARY\_COLOR\_ARRAY, 34  
 GL\_SECONDARY\_COLOR\_ARRAY\_BUFFER\_BINDING, 35  
 GL\_SECONDARY\_COLOR\_ARRAY\_POINTER, 34  
 GL\_SECONDARY\_COLOR\_ARRAY\_SIZE, 34  
 GL\_SECONDARY\_COLOR\_ARRAY\_STRIDE, 34  
 GL\_SECONDARY\_COLOR\_ARRAY\_TYPE, 34  
 GL\_SELECT, 21  
 GL\_SELECTION\_BUFFER\_POINTER, 21

GL_SELECTION_BUFFER_SIZE, 21	GL_STENCIL_BUFFER_BIT, 27
GL_SEPARABLE_2D, 30	GL_STENCIL_CLEAR_VALUE, 22
GL_SEPARATE_SPECULAR_COLOR, 29	GL_STENCIL_FAIL, 22
GL_SET, 22	GL_STENCIL_FUNC, 22
GL_SHADE_MODEL, 20	GL_STENCIL_INDEX, 22
GL_SHININESS, 20	GL_STENCIL_PASS_DEPTH_FAIL, 22
GL_SHORT, 17	GL_STENCIL_PASS_DEPTH_PASS, 22
GL_SINGLE_COLOR, 29	GL_STENCIL_REF, 22
GL_SMOOTH, 20	GL_STENCIL_TEST, 22
GL_SMOOTH_LINE_WIDTH_GRANULARITY, 29	GL_STENCIL_VALUE_MASK, 22
GL_SMOOTH_LINE_WIDTH_RANGE, 29	GL_STENCIL_WRITEMASK, 22
GL_SMOOTH_POINT_SIZE_GRANULARITY, 29	GL_STEREO, 23
GL_SMOOTH_POINT_SIZE_RANGE, 29	GL_STREAM_COPY, 35
GL_SOURCE0_ALPHA, 33	GL_STREAM_DRAW, 35
GL_SOURCE0_RGB, 33	GL_STREAM_READ, 35
GL_SOURCE1_ALPHA, 33	GL_SUBPIXEL_BITS, 23
GL_SOURCE1_RGB, 33	GL_SUBTRACT, 33
GL_SOURCE2_ALPHA, 33	GL_T, 27
GL_SOURCE2_RGB, 33	GL_T2F_C3F_V3F, 18
GL_SPECULAR, 20	GL_T2F_C4F_N3F_V3F, 18
GL_SPHERE_MAP, 27	GL_T2F_C4UB_V3F, 18
GL_SPOT_CUTOFF, 19	GL_T2F_N3F_V3F, 18
GL_SPOT_DIRECTION, 20	GL_T2F_V3F, 18
GL_SPOT_EXPONENT, 19	GL_T4F_C4F_N3F_V4F, 18
GL_SRC0_ALPHA, 35	GL_T4F_V4F, 18
GL_SRC0_RGB, 35	GL_TABLE_TOO_LARGE, 30
GL_SRC1_ALPHA, 35	GL_TEXTURE, 18
GL_SRC1_RGB, 35	GL_TEXTURE0, 31
GL_SRC2_ALPHA, 35	GL_TEXTURE1, 31
GL_SRC2_RGB, 35	GL_TEXTURE10, 31
GL_SRC_ALPHA, 21	GL_TEXTURE11, 31
GL_SRC_ALPHA_SATURATE, 21	GL_TEXTURE12, 31
GL_SRC_COLOR, 21	GL_TEXTURE13, 31
GL_STACK_OVERFLOW, 27	GL_TEXTURE14, 32
GL_STACK_UNDERFLOW, 27	GL_TEXTURE15, 32
GL_STATIC_COPY, 35	GL_TEXTURE16, 32
GL_STATIC_DRAW, 35	GL_TEXTURE17, 32
GL_STATIC_READ, 35	GL_TEXTURE18, 32
GL_STENCIL, 23	GL_TEXTURE19, 32
GL_STENCIL_BITS, 22	GL_TEXTURE2, 31
	GL_TEXTURE20, 32
	GL_TEXTURE21, 32

GL\_TEXTURE22, 32  
 GL\_TEXTURE23, 32  
 GL\_TEXTURE24, 32  
 GL\_TEXTURE25, 32  
 GL\_TEXTURE26, 32  
 GL\_TEXTURE27, 32  
 GL\_TEXTURE28, 32  
 GL\_TEXTURE29, 32  
 GL\_TEXTURE3, 31  
 GL\_TEXTURE30, 32  
 GL\_TEXTURE31, 32  
 GL\_TEXTURE4, 31  
 GL\_TEXTURE5, 31  
 GL\_TEXTURE6, 31  
 GL\_TEXTURE7, 31  
 GL\_TEXTURE8, 31  
 GL\_TEXTURE9, 31  
 GL\_TEXTURE\_1D, 26  
 GL\_TEXTURE\_2D, 26  
 GL\_TEXTURE\_3D, 29  
 GL\_TEXTURE\_ALPHA\_SIZE, 26  
 GL\_TEXTURE\_BASE\_LEVEL, 29  
 GL\_TEXTURE\_BINDING\_1D, 28  
 GL\_TEXTURE\_BINDING\_2D, 28  
 GL\_TEXTURE\_BINDING\_3D, 29  
 GL\_TEXTURE\_BINDING\_CUBE\_MAP, 32  
 GL\_TEXTURE\_BIT, 27  
 GL\_TEXTURE\_BLUE\_SIZE, 26  
 GL\_TEXTURE\_BORDER, 26  
 GL\_TEXTURE\_BORDER\_COLOR, 26  
 GL\_TEXTURE\_COMPARE\_FUNC, 34  
 GL\_TEXTURE\_COMPARE\_MODE, 34  
 GL\_TEXTURE\_COMPONENTS, 26  
 GL\_TEXTURE\_COMPRESSED, 33  
 GL\_TEXTURE\_COMPRESSED\_IMAGE\_SIZE,  
 33  
 GL\_TEXTURE\_COMPRESSION\_HINT, 33  
 GL\_TEXTURE\_COORD\_ARRAY, 17  
 GL\_TEXTURE\_COORD\_ARRAY\_BUFFER\_BINDING,  
 35  
 GL\_TEXTURE\_COORD\_ARRAY\_POINTER, 18  
 GL\_TEXTURE\_COORD\_ARRAY\_SIZE, 17  
 GL\_TEXTURE\_COORD\_ARRAY\_STRIDE, 17  
 GL\_TEXTURE\_COORD\_ARRAY\_TYPE, 17  
 GL\_TEXTURE\_CUBE\_MAP, 32  
 GL\_TEXTURE\_CUBE\_MAP\_NEGATIVE\_X, 32  
 GL\_TEXTURE\_CUBE\_MAP\_NEGATIVE\_Y, 32  
 GL\_TEXTURE\_CUBE\_MAP\_NEGATIVE\_Z, 32  
 GL\_TEXTURE\_CUBE\_MAP\_POSITIVE\_X, 32  
 GL\_TEXTURE\_CUBE\_MAP\_POSITIVE\_Y, 32  
 GL\_TEXTURE\_CUBE\_MAP\_POSITIVE\_Z, 32  
 GL\_TEXTURE\_DEPTH, 29  
 GL\_TEXTURE\_DEPTH\_SIZE, 34  
 GL\_TEXTURE\_ENV, 26  
 GL\_TEXTURE\_ENV\_COLOR, 26  
 GL\_TEXTURE\_ENV\_MODE, 26  
 GL\_TEXTURE\_FILTER\_CONTROL, 34  
 GL\_TEXTURE\_GEN\_MODE, 26  
 GL\_TEXTURE\_GEN\_Q, 27  
 GL\_TEXTURE\_GEN\_R, 27  
 GL\_TEXTURE\_GEN\_S, 26  
 GL\_TEXTURE\_GEN\_T, 26  
 GL\_TEXTURE\_GREEN\_SIZE, 26  
 GL\_TEXTURE\_HEIGHT, 26  
 GL\_TEXTURE\_INTENSITY\_SIZE, 26  
 GL\_TEXTURE\_INTERNAL\_FORMAT, 28  
 GL\_TEXTURE\_LOD\_BIAS, 34  
 GL\_TEXTURE\_LUMINANCE\_SIZE, 26  
 GL\_TEXTURE\_MAG\_FILTER, 26  
 GL\_TEXTURE\_MATRIX, 24  
 GL\_TEXTURE\_MAX\_LEVEL, 29  
 GL\_TEXTURE\_MAX\_LOD, 29  
 GL\_TEXTURE\_MIN\_FILTER, 26  
 GL\_TEXTURE\_MIN\_LOD, 29  
 GL\_TEXTURE\_PRIORITY, 28  
 GL\_TEXTURE\_RED\_SIZE, 26  
 GL\_TEXTURE\_RESIDENT, 28  
 GL\_TEXTURE\_STACK\_DEPTH, 24  
 GL\_TEXTURE\_WIDTH, 26  
 GL\_TEXTURE\_WRAP\_R, 29  
 GL\_TEXTURE\_WRAP\_S, 26  
 GL\_TEXTURE\_WRAP\_T, 26  
 GL\_TRANSFORM\_BIT, 27  
 GL\_TRANSPOSE\_COLOR\_MATRIX, 32



GL_TRANSPOSE_MODELVIEW_MATRIX, 32	35
GL_TRANSPOSE_PROJECTION_MATRIX, 32	GL_VIEWPORT, 24
GL_TRANSPOSE_TEXTURE_MATRIX, 32	GL_VIEWPORT_BIT, 27
GL_TRIANGLE_FAN, 17	GL_WEIGHT_ARRAY_BUFFER_BINDING, 35
GL_TRIANGLE_STRIP, 17	GL_WRITE_ONLY, 35
GL_TRIANGLES, 17	GL_XOR, 22
GL_TRUE, 17	GL_ZERO, 20
GL_TRUE, 4	GL_ZOOM_X, 26
GL_UNPACK_ALIGNMENT, 26	GL_ZOOM_Y, 26
GL_UNPACK_IMAGE_HEIGHT, 29	glAccum, 7
GL_UNPACK_LSB_FIRST, 26	glActiveTexture, 7
GL_UNPACK_ROW_LENGTH, 26	glAlphaFunc, 7
GL_UNPACK_SKIP_IMAGES, 29	glAreTexturesResident, 5
GL_UNPACK_SKIP_PIXELS, 26	glBegin, 7
GL_UNPACK_SKIP_ROWS, 26	glBeginQuery, 7
GL_UNPACK_SWAP_BYTES, 26	glBindTexture, 7
GL_UNSIGNED_BYTE, 17	GLbitfield, 4
GL_UNSIGNED_BYTE_2_3_3_REV, 29	glBitmap, 7
GL_UNSIGNED_BYTE_3_3_2, 29	glBlendColor, 7
GL_UNSIGNED_INT, 17	glBlendEquation, 7
GL_UNSIGNED_INT_10_10_10_2, 29	glBlendFunc, 7
GL_UNSIGNED_INT_2_10_10_10_REV, 29	glBlendFuncSeparate, 7
GL_UNSIGNED_INT_8_8_8_8, 29	GLboolean, 4
GL_UNSIGNED_INT_8_8_8_8_REV, 29	GLbyte, 4
GL_UNSIGNED_SHORT, 17	glCallList, 7
GL_UNSIGNED_SHORT_1_5_5_5_REV, 29	glCallLists, 7
GL_UNSIGNED_SHORT_4_4_4_4, 29	GLclampd, 4
GL_UNSIGNED_SHORT_4_4_4_4_REV, 29	glClear, 7
GL_UNSIGNED_SHORT_5_5_5_1, 29	glClearAccum, 7
GL_UNSIGNED_SHORT_5_6_5, 29	glClearColor, 7
GL_UNSIGNED_SHORT_5_6_5_REV, 29	glClearDepth, 7
GL_V2F, 18	glClearIndex, 7
GL_V3F, 18	glClearStencil, 7
GL_VENDOR, 27	glClipPlane, 7
GL_VERSION, 27	glColor3b, 7
GL_VERTEX_ARRAY, 17	glColor3bv, 7
GL_VERTEX_ARRAY_BUFFER_BINDING, 34	glColor3d, 7
GL_VERTEX_ARRAY_POINTER, 18	glColor3dv, 7
GL_VERTEX_ARRAY_SIZE, 17	glColor3f, 7
GL_VERTEX_ARRAY_STRIDE, 17	glColor3fv, 7
GL_VERTEX_ARRAY_TYPE, 17	glColor3i, 7
GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING, 35	glColor3iv, 7

glColor3s, 7  
 glColor3sv, 7  
 glColor3ub, 7  
 glColor3ubv, 7  
 glColor3ui, 8  
 glColor3uiv, 8  
 glColor3us, 8  
 glColor3usv, 8  
 glColor4b, 8  
 glColor4bv, 8  
 glColor4d, 8  
 glColor4dv, 8  
 glColor4f, 8  
 glColor4fv, 8  
 glColor4i, 8  
 glColor4iv, 8  
 glColor4s, 8  
 glColor4sv, 8  
 glColor4ub, 8  
 glColor4ubv, 8  
 glColor4ui, 8  
 glColor4uiv, 8  
 glColor4us, 8  
 glColor4usv, 8  
 glColorMask, 8  
 glColorMaterial, 8  
 glColorSubTable, 8  
 glColorTable, 8  
 glColorTableParameterfv, 8  
 glColorTableParameteriv, 8  
 glCompressedTexImage1D, 8  
 glCompressedTexImage2D, 8  
 glCompressedTexImage3D, 8  
 glCompressedTexSubImage1D, 8  
 glCompressedTexSubImage2D, 8  
 glCompressedTexSubImage3D, 8  
 glConvolutionFilter1D, 8  
 glConvolutionFilter2D, 8  
 glConvolutionParameterf, 8  
 glConvolutionParameterfv, 8  
 glConvolutionParameteri, 8  
 glConvolutionParameteriv, 8  
 glCopyColorSubTable, 8  
 glCopyColorTable, 8  
 glCopyConvolutionFilter1D, 8  
 glCopyConvolutionFilter2D, 8  
 glCopyPixels, 8  
 glCopyTexImage1D, 8  
 glCopyTexImage2D, 9  
 glCopyTexSubImage1D, 9  
 glCopyTexSubImage2D, 9  
 glCopyTexSubImage3D, 9  
 glCullFace, 9  
 glDeleteLists, 9  
 glDeleteQueries, 5  
 glDeleteTextures, 5  
 glDepthFunc, 9  
 glDepthMask, 9  
 glDepthRange, 9  
 glDisable, 9  
 GLdouble, 4  
 glDrawBuffer, 9  
 glDrawPixels, 9  
 glEdgeFlag, 9  
 glEdgeFlagv, 9  
 glEnable, 9  
 glEnd, 9  
 glEndList, 9  
 glEndQuery, 9  
 GLenum, 4  
 glEvalCoord1d, 9  
 glEvalCoord1dv, 9  
 glEvalCoord1f, 9  
 glEvalCoord1fv, 9  
 glEvalCoord2d, 9  
 glEvalCoord2dv, 9  
 glEvalCoord2f, 9  
 glEvalCoord2fv, 9  
 glEvalMesh1, 9  
 glEvalMesh2, 9  
 glEvalPoint1, 9  
 glEvalPoint2, 9  
 glFeedbackBuffer, 7  
 glFinish, 9

glFlush, 9  
 glFogCoorddd, 9  
 glFogCoorddv, 9  
 glFogCoordf, 9  
 glFogCoordfv, 9  
 glFogf, 9  
 glFogfv, 9  
 glFogi, 9  
 glFogiv, 9  
 glFrontFace, 9  
 glFrustum, 9  
 glGenLists, 9  
 glGenQueries, 5  
 glGenTextures, 5  
 glGetBooleanv, 5  
 glGetBufferParameteriv, 6  
 glGetClipPlane, 6  
 glGetColorTable, 9  
 glGetCompressedTexImage, 10  
 glGetConvolutionFilter, 10  
 glGetConvolutionParameterfv, 6  
 glGetConvolutionParameteriv, 6  
 glGetDoublev, 5  
 glGetError, 10  
 glGetFloatv, 5  
 glGetHistogram, 10  
 glGetHistogramParameterfv, 6  
 glGetHistogramParameteriv, 6  
 glGetIntegerv, 5  
 glGetLightfv, 5  
 glGetLightiv, 5  
 glGetMapdv, 6  
 glGetMapfv, 6  
 glGetMapiv, 6  
 glGetMaterialfv, 5  
 glGetMaterialiv, 5  
 glGetMinmax, 10  
 glGetMinmaxParameterfv, 6  
 glGetMinmaxParameteriv, 6  
 glGetPixelMapfv, 6  
 glGetPixelMapuiv, 6  
 glGetPixelMapusv, 6  
 glGetPolygonStipple, 10  
 glGetQueryiv, 6  
 glGetQueryObjectiv, 6  
 glGetQueryObjectuiv, 6  
 glGetSeparableFilter, 10  
 getString, 6  
 glGetTexEnvfv, 5  
 glGetTexEnviv, 5  
 glGetTexGendv, 6  
 glGetTexGenfv, 6  
 glGetTexGeniv, 6  
 glGetTexImage, 10  
 glGetTexLevelParameterfv, 6  
 glGetTexLevelParameteriv, 6  
 glGetTexParameterfv, 6  
 glGetTexParameteriv, 6  
 glHint, 10  
 glHistogram, 10  
 glIndexd, 10  
 glIndexdv, 10  
 glIndexf, 10  
 glIndexfv, 10  
 glIndexi, 10  
 glIndexiv, 10  
 glIndexMask, 10  
 glIndexs, 10  
 glIndexsv, 10  
 glIndexub, 10  
 glIndexubv, 10  
 glInitNames, 10  
 GLint, 4  
 glIsBuffer, 10  
 glIsEnabled, 10  
 glIsList, 10  
 glIsQuery, 10  
 glIsTexture, 10  
 glLightf, 10  
 glLightfv, 10  
 glLighti, 10  
 glLightiv, 10  
 glLightModelf, 10  
 glLightModelfv, 10

glLightModeli, 10  
 glLightModeliv, 10  
 glLineStipple, 10  
 glLineWidth, 10  
 glListBase, 10  
 glLoadIdentity, 10  
 glLoadMatrixd, 10  
 glLoadMatrixf, 10  
 glLoadName, 10  
 glLoadTransposeMatrixd, 10  
 glLoadTransposeMatrixf, 10  
 glLogicOp, 11  
 glMap1d, 11  
 glMap1f, 11  
 glMap2d, 11  
 glMap2f, 11  
 glMapGrid1d, 11  
 glMapGrid1f, 11  
 glMapGrid2d, 11  
 glMapGrid2f, 11  
 glMaterialf, 11  
 glMaterialfv, 11  
 glMateriali, 11  
 glMaterialiv, 11  
 glMatrixMode, 11  
 glMinmax, 11  
 glMultiTexCoord1d, 11  
 glMultiTexCoord1dv, 11  
 glMultiTexCoord1f, 11  
 glMultiTexCoord1fv, 11  
 glMultiTexCoord1i, 11  
 glMultiTexCoord1iv, 11  
 glMultiTexCoord1s, 11  
 glMultiTexCoord1sv, 11  
 glMultiTexCoord2d, 11  
 glMultiTexCoord2dv, 11  
 glMultiTexCoord2f, 11  
 glMultiTexCoord2fv, 11  
 glMultiTexCoord2i, 11  
 glMultiTexCoord2iv, 11  
 glMultiTexCoord2s, 11  
 glMultiTexCoord2sv, 11  
 glMultiTexCoord3d, 11  
 glMultiTexCoord3dv, 11  
 glMultiTexCoord3f, 11  
 glMultiTexCoord3fv, 11  
 glMultiTexCoord3i, 11  
 glMultiTexCoord3iv, 11  
 glMultiTexCoord3s, 11  
 glMultiTexCoord3sv, 11  
 glMultiTexCoord4d, 11  
 glMultiTexCoord4dv, 12  
 glMultiTexCoord4f, 12  
 glMultiTexCoord4fv, 12  
 glMultiTexCoord4i, 12  
 glMultiTexCoord4iv, 12  
 glMultiTexCoord4s, 12  
 glMultiTexCoord4sv, 12  
 glMultMatrixd, 11  
 glMultMatrixf, 11  
 glMultTransposeMatrixd, 11  
 glMultTransposeMatrixf, 11  
 glNewList, 12  
 glNormal3b, 12  
 glNormal3bv, 12  
 glNormal3d, 12  
 glNormal3dv, 12  
 glNormal3f, 12  
 glNormal3fv, 12  
 glNormal3i, 12  
 glNormal3iv, 12  
 glNormal3s, 12  
 glNormal3sv, 12  
 glOrtho, 12  
 glPassThrough, 12  
 glPixelMapfv, 5  
 glPixelMapuiv, 5  
 glPixelMapusv, 5  
 glPixelStoref, 12  
 glPixelStorei, 12  
 glPixelTransferf, 12  
 glPixelTransferi, 12  
 glPixelZoom, 12  
 glPointParameterf, 12

[glPointParameterfv](#), 12  
[glPointParameteri](#), 12  
[glPointParameteriv](#), 12  
[glPointSize](#), 12  
[glPolygonMode](#), 12  
[glPolygonOffset](#), 12  
[glPolygonStipple](#), 12  
[glPopAttrib](#), 12  
[glPopClientAttrib](#), 12  
[glPopMatrix](#), 12  
[glPopName](#), 12  
[glPushAttrib](#), 12  
[glPushClientAttrib](#), 12  
[glPushMatrix](#), 12  
[glPushName](#), 12  
[glRasterPos2d](#), 12  
[glRasterPos2dv](#), 12  
[glRasterPos2f](#), 12  
[glRasterPos2fv](#), 13  
[glRasterPos2i](#), 13  
[glRasterPos2iv](#), 13  
[glRasterPos2s](#), 13  
[glRasterPos2sv](#), 13  
[glRasterPos3d](#), 13  
[glRasterPos3dv](#), 13  
[glRasterPos3f](#), 13  
[glRasterPos3fv](#), 13  
[glRasterPos3i](#), 13  
[glRasterPos3iv](#), 13  
[glRasterPos3s](#), 13  
[glRasterPos3sv](#), 13  
[glRasterPos4d](#), 13  
[glRasterPos4dv](#), 13  
[glRasterPos4f](#), 13  
[glRasterPos4fv](#), 13  
[glRasterPos4i](#), 13  
[glRasterPos4iv](#), 13  
[glRasterPos4s](#), 13  
[glRasterPos4sv](#), 13  
[glReadBuffer](#), 13  
[glReadPixels](#), 13  
[glRectd](#), 13  
[glRectdv](#), 13  
[glRectf](#), 13  
[glRectfv](#), 13  
[glRecti](#), 13  
[glRectiv](#), 13  
[glRects](#), 13  
[glRectsv](#), 13  
[glRenderMode](#), 13  
[glResetHistogram](#), 13  
[glResetMinmax](#), 13  
[glRotated](#), 13  
[glRotatef](#), 13  
[glSampleCoverage](#), 13  
[glScaled](#), 13  
[glScalef](#), 13  
[glScissor](#), 13  
[glSecondaryColor3b](#), 13  
[glSecondaryColor3bv](#), 13  
[glSecondaryColor3d](#), 13  
[glSecondaryColor3dv](#), 13  
[glSecondaryColor3f](#), 14  
[glSecondaryColor3fv](#), 14  
[glSecondaryColor3i](#), 14  
[glSecondaryColor3iv](#), 14  
[glSecondaryColor3s](#), 14  
[glSecondaryColor3sv](#), 14  
[glSecondaryColor3ub](#), 14  
[glSecondaryColor3ubv](#), 14  
[glSecondaryColor3ui](#), 14  
[glSecondaryColor3uiv](#), 14  
[glSecondaryColor3us](#), 14  
[glSecondaryColor3usv](#), 14  
[glSelectBuffer](#), 7  
[glSeparableFilter2D](#), 14  
[glShadeModel](#), 14  
[GLshort](#), 4  
[GLsizei](#), 4  
[glStencilFunc](#), 14  
[glStencilMask](#), 14  
[glStencilOp](#), 14  
[glTexCoord1d](#), 14  
[glTexCoord1dv](#), 14

glTexCoord1f, 14  
 glTexCoord1fv, 14  
 glTexCoord1i, 14  
 glTexCoord1iv, 14  
 glTexCoord1s, 14  
 glTexCoord1sv, 14  
 glTexCoord2d, 14  
 glTexCoord2dv, 14  
 glTexCoord2f, 14  
 glTexCoord2fv, 14  
 glTexCoord2i, 14  
 glTexCoord2iv, 14  
 glTexCoord2s, 14  
 glTexCoord2sv, 14  
 glTexCoord3d, 14  
 glTexCoord3dv, 14  
 glTexCoord3f, 14  
 glTexCoord3fv, 14  
 glTexCoord3i, 14  
 glTexCoord3iv, 14  
 glTexCoord3s, 14  
 glTexCoord3sv, 14  
 glTexCoord4d, 14  
 glTexCoord4dv, 14  
 glTexCoord4f, 14  
 glTexCoord4fv, 15  
 glTexCoord4i, 15  
 glTexCoord4iv, 15  
 glTexCoord4s, 15  
 glTexCoord4sv, 15  
 glTexEnvf, 15  
 glTexEnvfv, 15  
 glTexEnvi, 15  
 glTexEnviv, 15  
 glTexGend, 15  
 glTexGendv, 15  
 glTexGenf, 15  
 glTexGenfv, 15  
 glTexGeni, 15  
 glTexGeniv, 15  
 glTexImage1D, 15  
 glTexImage2D, 15  
 glTexImage3D, 15  
 glTexParameterf, 15  
 glTexParameterfv, 15  
 glTexParameteri, 15  
 glTexParameteriv, 15  
 glTexSubImage1D, 15  
 glTexSubImage2D, 15  
 glTexSubImage3D, 15  
 glTranslated, 15  
 glTranslatef, 15  
 GLU, 1  
 GLU\_AUTO\_LOAD\_MATRIX, 37  
 GLU\_BEGIN, 38  
 GLU\_CCW, 38  
 GLU\_CULLING, 37  
 GLU\_CW, 38  
 GLU\_DISPLAY\_MODE, 37  
 GLU\_DOMAIN\_DISTANCE, 38  
 GLU\_EDGE\_FLAG, 38  
 GLU\_END, 38  
 GLU\_ERROR, 36  
 GLU\_EXTENSIONS, 35  
 GLU\_EXTERIOR, 38  
 GLU\_FALSE, 35  
 GLU\_FILL, 38  
 GLU\_FLAT, 38  
 GLU\_INSIDE, 38  
 GLU\_INTERIOR, 38  
 GLU\_INVALID\_ENUM, 35  
 GLU\_INVALID\_OPERATION, 36  
 GLU\_INVALID\_VALUE, 36  
 GLU\_LINE, 38  
 GLU\_MAP1\_TRIM\_2, 38  
 GLU\_MAP1\_TRIM\_3, 38  
 GLU\_NONE, 38  
 GLU\_NURBS\_BEGIN, 36  
 GLU\_NURBS\_BEGIN\_DATA, 36  
 GLU\_NURBS\_BEGIN\_DATA\_EXT, 36  
 GLU\_NURBS\_BEGIN\_EXT, 36  
 GLU\_NURBS\_COLOR, 36  
 GLU\_NURBS\_COLOR\_DATA, 36  
 GLU\_NURBS\_COLOR\_DATA\_EXT, 36

GLU\_NURBS\_COLOR\_EXT, 36  
 GLU\_NURBS\_END, 36  
 GLU\_NURBS\_END\_DATA, 36  
 GLU\_NURBS\_END\_DATA\_EXT, 36  
 GLU\_NURBS\_END\_EXT, 36  
 GLU\_NURBS\_ERROR, 36  
 GLU\_NURBS\_ERROR1, 36  
 GLU\_NURBS\_ERROR10, 36  
 GLU\_NURBS\_ERROR11, 36  
 GLU\_NURBS\_ERROR12, 36  
 GLU\_NURBS\_ERROR13, 36  
 GLU\_NURBS\_ERROR14, 37  
 GLU\_NURBS\_ERROR15, 37  
 GLU\_NURBS\_ERROR16, 37  
 GLU\_NURBS\_ERROR17, 37  
 GLU\_NURBS\_ERROR18, 37  
 GLU\_NURBS\_ERROR19, 37  
 GLU\_NURBS\_ERROR2, 36  
 GLU\_NURBS\_ERROR20, 37  
 GLU\_NURBS\_ERROR21, 37  
 GLU\_NURBS\_ERROR22, 37  
 GLU\_NURBS\_ERROR23, 37  
 GLU\_NURBS\_ERROR24, 37  
 GLU\_NURBS\_ERROR25, 37  
 GLU\_NURBS\_ERROR26, 37  
 GLU\_NURBS\_ERROR27, 37  
 GLU\_NURBS\_ERROR28, 37  
 GLU\_NURBS\_ERROR29, 37  
 GLU\_NURBS\_ERROR3, 36  
 GLU\_NURBS\_ERROR30, 37  
 GLU\_NURBS\_ERROR31, 37  
 GLU\_NURBS\_ERROR32, 37  
 GLU\_NURBS\_ERROR33, 37  
 GLU\_NURBS\_ERROR34, 37  
 GLU\_NURBS\_ERROR35, 37  
 GLU\_NURBS\_ERROR36, 37  
 GLU\_NURBS\_ERROR37, 37  
 GLU\_NURBS\_ERROR4, 36  
 GLU\_NURBS\_ERROR5, 36  
 GLU\_NURBS\_ERROR6, 36  
 GLU\_NURBS\_ERROR7, 36  
 GLU\_NURBS\_ERROR8, 36  
 GLU\_NURBS\_ERROR9, 36  
 GLU\_NURBS\_MODE, 37  
 GLU\_NURBS\_MODE\_EXT, 37  
 GLU\_NURBS\_NORMAL, 36  
 GLU\_NURBS\_NORMAL\_DATA, 36  
 GLU\_NURBS\_NORMAL\_DATA\_EXT, 36  
 GLU\_NURBS\_NORMAL\_EXT, 36  
 GLU\_NURBS\_RENDERER, 37  
 GLU\_NURBS\_RENDERER\_EXT, 37  
 GLU\_NURBS\_TESSELLATOR, 37  
 GLU\_NURBS\_TESSELLATOR\_EXT, 37  
 GLU\_NURBS\_TEX\_COORD\_DATA\_EXT, 36  
 GLU\_NURBS\_TEX\_COORD\_EXT, 36  
 GLU\_NURBS\_TEXTURE\_COORD, 36  
 GLU\_NURBS\_TEXTURE\_COORD\_DATA, 36  
 GLU\_NURBS\_VERTEX, 36  
 GLU\_NURBS\_VERTEX\_DATA, 36  
 GLU\_NURBS\_VERTEX\_DATA\_EXT, 36  
 GLU\_NURBS\_VERTEX\_EXT, 36  
 GLU\_OBJECT\_PARAMETRIC\_ERROR, 37  
 GLU\_OBJECT\_PARAMETRIC\_ERROR\_EXT, 37  
 GLU\_OBJECT\_PATH\_LENGTH, 37  
 GLU\_OBJECT\_PATH\_LENGTH\_EXT, 37  
 GLU\_OUT\_OF\_MEMORY, 36  
 GLU\_OUTLINE\_PATCH, 36  
 GLU\_OUTLINE\_POLYGON, 36  
 GLU\_OUTSIDE, 38  
 GLU\_PARAMETRIC\_ERROR, 37  
 GLU\_PARAMETRIC\_TOLERANCE, 37  
 GLU\_PATH\_LENGTH, 37  
 GLU\_POINT, 38  
 GLU\_SAMPLING\_METHOD, 37  
 GLU\_SAMPLING\_TOLERANCE, 37  
 GLU\_SILHOUETTE, 38  
 GLU\_SMOOTH, 38  
 GLU\_TESS\_BEGIN, 38  
 GLU\_TESS\_BEGIN\_DATA, 38  
 GLU\_TESS\_BOUNDARY\_ONLY, 38  
 GLU\_TESS\_COMBINE, 38  
 GLU\_TESS\_COMBINE\_DATA, 38  
 GLU\_TESS\_COORD\_TOO\_LARGE, 39  
 GLU\_TESS\_EDGE\_FLAG, 38

GLU\_TESS\_EDGE\_FLAG\_DATA, 38  
 GLU\_TESS\_END, 38  
 GLU\_TESS\_END\_DATA, 38  
 GLU\_TESS\_ERROR, 38  
 GLU\_TESS\_ERROR1, 38  
 GLU\_TESS\_ERROR2, 38  
 GLU\_TESS\_ERROR3, 38  
 GLU\_TESS\_ERROR4, 38  
 GLU\_TESS\_ERROR5, 38  
 GLU\_TESS\_ERROR6, 38  
 GLU\_TESS\_ERROR7, 38  
 GLU\_TESS\_ERROR8, 38  
 GLU\_TESS\_ERROR\_DATA, 38  
 GLU\_TESS\_MAX\_COORD, 39  
 GLU\_TESS\_MISSING\_BEGIN\_CONTOUR, 39  
 GLU\_TESS\_MISSING\_BEGIN\_POLYGON, 39  
 GLU\_TESS\_MISSING\_END\_CONTOUR, 39  
 GLU\_TESS\_MISSING\_END\_POLYGON, 39  
 GLU\_TESS\_NEED\_COMBINE\_CALLBACK, 39  
 GLU\_TESS\_TOLERANCE, 38  
 GLU\_TESS\_VERTEX, 38  
 GLU\_TESS\_VERTEX\_DATA, 38  
 GLU\_TESS\_WINDING\_ABS\_GEQ\_TWO, 39  
 GLU\_TESS\_WINDING\_NEGATIVE, 39  
 GLU\_TESS\_WINDING\_NONZERO, 39  
 GLU\_TESS\_WINDING\_ODD, 39  
 GLU\_TESS\_WINDING\_POSITIVE, 39  
 GLU\_TESS\_WINDING\_RULE, 38  
 GLU\_TRUE, 35  
 GLU\_U\_STEP, 37  
 GLU\_UNKNOWN, 38  
 GLU\_V\_STEP, 37  
 GLU\_VERSION, 35  
 GLU\_VERTEX, 38  
 gluBuild1DMipmapLevels, 16  
 gluBuild1DMipmaps, 16  
 gluBuild2DMipmapLevels, 16  
 gluBuild2DMipmaps, 16  
 gluBuild3DMipmapLevels, 16  
 gluBuild3DMipmaps, 16  
 GLubyte, 4  
 gluCheckExtension, 6  
 gluCylinder, 16  
 gluDisk, 16  
 gluErrorString, 6  
 gluGetString, 6  
 GLuint, 4  
 gluLookAt, 16  
 gluNewQuadric, 16  
 gluOrtho2D, 16  
 gluPartialDisk, 16  
 gluPerspective, 16  
 gluPickMatrix, 16  
 gluProject, 6  
 gluQuadricDrawStyle, 16  
 gluQuadricNormals, 16  
 gluQuadricOrientation, 16  
 gluQuadricTexture, 16  
 gluScaleImage, 16  
 GLushort, 4  
 gluSphere, 16  
 gluUnProject, 6  
 gluUnProject4, 6  
 glVertex2d, 15  
 glVertex2dv, 15  
 glVertex2f, 15  
 glVertex2fv, 15  
 glVertex2i, 15  
 glVertex2iv, 15  
 glVertex2s, 15  
 glVertex2sv, 15  
 glVertex3d, 15  
 glVertex3dv, 15  
 glVertex3f, 15  
 glVertex3fv, 15  
 glVertex3i, 15  
 glVertex3iv, 15  
 glVertex3s, 15  
 glVertex3sv, 15  
 glVertex4d, 15  
 glVertex4dv, 16  
 glVertex4f, 16  
 glVertex4fv, 16  
 glVertex4i, 16



- [glVertex4iv](#), 16
- [glVertex4s](#), 16
- [glVertex4sv](#), 16
- [glViewport](#), 16
- [glWindowPos2d](#), 16
- [glWindowPos2dv](#), 16
- [glWindowPos2f](#), 16
- [glWindowPos2fv](#), 16
- [glWindowPos2i](#), 16
- [glWindowPos2iv](#), 16
- [glWindowPos2s](#), 16
- [glWindowPos2sv](#), 16
- [glWindowPos3d](#), 16
- [glWindowPos3dv](#), 16
- [glWindowPos3f](#), 16
- [glWindowPos3fv](#), 16
- [glWindowPos3i](#), 16
- [glWindowPos3iv](#), 16
- [glWindowPos3s](#), 16
- [glWindowPos3sv](#), 16
- Initialization, 52
- [list->gl-boolean-vector](#), 49
- [list->gl-byte-vector](#), 45
- [list->gl-double-vector](#), 49
- [list->gl-float-vector](#), 48
- [list->gl-int-vector](#), 47
- [list->gl-short-vector](#), 46
- [list->gl-ubyte-vector](#), 46
- [list->gl-uint-vector](#), 48
- [list->gl-ushort-vector](#), 47
- [make-gl-boolean-vector](#), 49
- [make-gl-byte-vector](#), 45
- [make-gl-double-vector](#), 49
- [make-gl-float-vector](#), 48
- [make-gl-int-vector](#), 47
- [make-gl-selection-record](#), 40
- [make-gl-short-vector](#), 46
- [make-gl-ubyte-vector](#), 46
- [make-gl-uint-vector](#), 48
- [make-gl-ushort-vector](#), 47
- OpenGL, 1
- OpenGL Vectors, 45
- Racket-Style OpenGL, 40
- [select-buffer->gl-uint-vector](#), 39
- [sgl](#), 40
- [sgl/bitmap](#), 51
- [sgl/gl](#), 4
- [sgl/gl-vectors](#), 45
- [sgl/init](#), 52
- [struct:gl-selection-record](#), 40
- Using OpenGL, 3
- [vector->gl-boolean-vector](#), 49
- [vector->gl-byte-vector](#), 45
- [vector->gl-double-vector](#), 49
- [vector->gl-float-vector](#), 48
- [vector->gl-int-vector](#), 47
- [vector->gl-short-vector](#), 46
- [vector->gl-ubyte-vector](#), 46
- [vector->gl-uint-vector](#), 48
- [vector->gl-ushort-vector](#), 47