



Index

Numbers

3D Computer Graphics, 360

A

accumulation buffer, 454, 456, 473–486
 clearing, 32, 457
 depth-of-field effect, use for, 481–485
 examples of use, 473
 full range for best results, use, 704
 motion blur, use for, 480
 sample program with depth-of-field effect, 483
 sample program with full-scene antialiasing, 475
 scene antialiasing, use for, 474
AGL, 14
airbrushing, 611
Akeley, Kurt, 473
aliasing, *See* antialiasing
alpha, 225
 destination alpha, 254
 material properties, 207
 multisampling coverage, 253
 texture image data type, 411
alpha blending, *See* blending
alpha test, 462
 querying current values, 463
 rendering pipeline stage, 14, 617
ambient
 contribution to lighting equation, 217
 global light, 202, 216
 light, 181, 182, 190
 material properties, 183, 207
animation, 21–24, 704
antialiasing, 241–254
 accumulation buffer used for, 474–480
 characters (by masking), 595
 characters (by texturing), 606

 color-index mode, 246
 coverage values, 241
 enabling for points or lines, 243
 enabling for polygons, 254
 lines, 241, 243–249
 lines (by texturing), 606
 points, 243–249, 598
 polygons, 254
 RGBA mode, 243
 sample program in color-index mode, 246
 sample program in RGBA mode, 244
 scene, with the accumulation buffer, 474
Apple Interface to OpenGL, *See* AGL
ARB imaging subset, 336–358
architectural applications
 orthographic parallel projection, use of, 131
Architecture Review Board
 extensions, approved, 587
arcs, 506
array elements, *See* vertex arrays
aspect ratio
 perspective projection, 129
 viewport transformation, 134
atmospheric effects, *See* fog
attenuation of light, 191–192
attribute groups, 83–86
 client, 84
 list of, 621–657
 multitexturing, with, 435
 performance tips, 706
 server, 84
 stack depth, obtaining, 84
 stacks, 84
auxiliary buffers, 455, 458

B

back-facing polygons, 57
 culling, 58



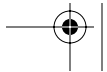


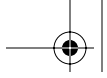
back-facing polygons (*Continued*)
 material property, specifying, 206
 two-sided lighting, 203
background, 30–32
 color, 30
 drawing a fixed, 460, 607
background processing, 689
backward compatibility
 tessellation, 504
basis functions, 517, 518
Bernstein
 basis, 517
 polynomial, 521
Bézier
 basis, 517, 518
 curve, 521
 sample program using mesh for surface, 528
 sample program which draws curve, 519
 sample program which draws surface, 526
BGR and BGRA pixel formats, 309
billboarding, 233, 463
bitmaps, 296–302
 display lists cache bitmap data, 277
 distorting, 593
 drawing, 300
 feedback mode, 575
 fonts, used for, 298, 304
 imaging pipeline operations, 317
 ordering of data in, 299
 origin of, 301
 sample program, 298
 sample program that creates a font, 305
 size of, 299
bitplanes, 164, 452
 displayable colors, number of, 166
blending, 225–238, 470
 antialiasing polygons, 254
 coverage calculations for antialiasing, 241
 destination alpha, 254
 differences among releases, 224
 enabling, 229
 enabling for antialiasing, 243
 equation, 229
 factors (source and destination), 226

images, 598
ordering polygons before drawing, 237
rendering pipeline stage, 14, 617
sample program for three-dimensional, 238
sample program with blended polygons, 235
texture function, 414
three dimensions, in, 237
uses of, 232
buffer, *See* framebuffer

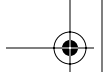
C

C programming language, 8
CAD/CAM, *See* computer-aided design
camera analogy, 100–101
 environment mapping, 428
 viewport transformations, 133
capping, *See* computational solid geometry
Chapter, 48
characters
 antialiasing, 606
circles, 506
clearing the framebuffer, 30–32, 456–457
 affected by scissoring, dithering, and masking, 457, 618
 performance tips, 707
client-server, *See* networked operation
clip coordinates, 102, 145
 feedback mode, 575
clipping, 133
 interference regions found using clipping planes, 602
 overview, 99
 primitives in rendering pipeline, 12
 viewing volume, 128
clipping planes
 additional clipping planes, 102, 144–147
 depth-buffer resolution, effect on, 704
 far, 129–132, 137
 near, 129–132, 137
 querying number of additional, 145
 sample program with additional clipping



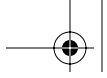


- planes, 146
- color
 - alpha values, 225
 - background, 30
 - cube showing blended RGB values, 163
 - current raster color, 302
 - human perception, 161
 - RGBA values for, 33, 162
 - specifying, 33
 - specifying for tessellation, 494
 - specifying in color-index mode, 172
 - specifying in RGBA mode, 171
- color buffer, 162, 164, 453, 454, 455
 - clearing, 32
 - masking, 459
- color map, 162, 167
 - loading for antialiasing, 246
 - loading for smooth shading, 175
 - loading, using GLUT, 687
 - size of, 168
- color matrix, 351–352
 - example, 351
 - post transform scale and bias, 352
 - sample program, 352
- color sum mode, 444
- color tables, 338–340
 - proxies, 343
 - replacing part of, 342
 - sample program, 340
 - specifying, 338
- color-index mode, 167–169
 - changing between RGBA mode and, 170
 - choosing between RGBA mode and, 169
 - coverage calculations for antialiasing, 241
 - dithering, 471
 - layering with writemasks, 460
 - lighting, 220–222
 - lighting calculations in, 221
 - texturing limitations, 365, 372
 - vertex arrays, specifying values with, 70
- combiner functions, 437–443
- command syntax, 7–9
- compositing images, 233
- compositing transformations, 147–154
- compressed texture images, 386–388
- computational solid geometry
 - capping, 469
 - difference of several contours, 499
 - interference regions, 601
 - intersection of two contours, 499
 - union of several contours, 499
- Computer Graphics: Principles and Practice*, xxi
- Computer Graphics: Principles and Practice*, 161, 165, 697
- computer-aided design
 - orthographic parallel projection, use of, 131
- concave polygons
 - GLU tessellation, 488
 - stencil buffer, drawing with the, 600
- cones, 506, 688
 - improving rendering of, 609
- constant attenuation, 192
- contours, 423
- control points, 516, 520, 524, 533
- convex polygons, 39
- convolutions, 343–??
 - 1d filters, 349
 - 2d filters, 343
 - border modes, 350
 - post convolution scale and bias, 351
 - sample program, 345
 - separable filters, 347
- Conway, John, 609
- coordinate systems
 - grand, fixed, 113, 122, 148
 - local, 113, 122, 148, 152
 - simple 2D, 36
- coordinates, *See* clip coordinates, depth
 - coordinates, eye coordinates,
 - homogeneous coordinates, normalized
 - device coordinates, object coordinates, q
 - texture coordinates, texture coordinates,
 - w coordinates, or window coordinates
- coverage, pixel, 241
- Coxeter, H. S. M., 697
- cross product, 125, 693
- CSG, *See* computational solid geometry



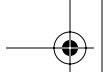
- cube maps, 430
- culling, 57–58
 - enabling, 58
 - rendering pipeline stage, 12, 616
 - selection mode, 556
- curves and curved surfaces, 41
 - see also* evaluators or NURBS
- Curves and Surfaces for Computer-Aided Geometric Design*, 517
- cylinders, 506
- D**
- data types
 - RGBA color conversion, 171
 - special OpenGL, 8
 - texture data, 371
 - warning about data type conversions, 705
- decals, 463, 599
 - polygon offset used for, 268
 - texture function, 413
- depth buffer, 180, 454, 455
 - also see* hidden-surface removal
 - background, using masking for a common, 460
 - blending, use for three-dimensional, 237
 - clearing, 32, 180, 457
 - decals, for, 599
 - Dirichlet domains, for, 609
 - drawing static backgrounds, 607
 - masking, 459
 - near frustum plane effect on resolution, 704
 - pixel data, 315, 326
- depth coordinates, 103, 135
- perspective division, 136
- picking use, 567
- polygon offset, 268–270
- rendering pipeline stage for depth-range operations, 12, 616
- sample program with picking, 567
- selection hit records, 556
- depth test, 469
 - also see* depth buffer
- rendering pipeline stage, 14, 617
- depth-cuing, *See* fog
- depth-of-field effect, 481–485
 - sample program, 483
- destination factor, *See* blending
- diffuse
 - contribution to lighting equation, 217
 - light, 182, 190
 - material properties, 183, 207
- directional light source, 191
- Dirichlet domains, 608
- disks, 506
- display lists, 29, 273
 - changing mode settings, 293
 - compiling, 281
 - creating, 279
 - deleting, 286
 - disadvantages, 278, 284
 - error handling, 280, 704
 - executing, 279, 283
 - executing multiple, 286
 - font creation, 287, 303
 - hierarchical, 284
 - immediate mode, mixing with, 283
 - indices for, obtaining, 280
 - naming, 280
 - nesting, 284
 - nesting limit, querying, 284
 - networked operation, 283
 - performance tips, 706
 - querying use of an index, 285
 - rendering pipeline stage, 11
 - sample program creating a font, 288
 - sample program for creating, 273, 279
 - sharing among rendering contexts, 663, 678
 - state variables saved and restored, 293
 - tessellation, use with, 503
 - uses for, 277, 293
 - vertex-array data, 283
 - what can be stored in, 282
- distorted images, 593
 - texture images, 416
- dithering, 166–167, 471, 704
 - and clearing, 457





- rendering pipeline stage, 14, 617
- dot product
 - lighting calculations, use in, 217
 - texture combiner function, 440
- double-buffering, 22–24
 - automatic glFlush(), 35
 - changing between single-buffering and, 170
 - object selection using the back buffer, 592
 - querying its presence, 455
 - sample program, 24
- drawing
 - clearing the window, 30
 - forcing completion of, 34
 - icosahedron, 88
 - points, 43
 - polygons, 43, 57
 - preparing for, 29
 - rectangles, 40
 - spheres, cylinders, and disks, 505–514
- drawing pixel data, *See* pixel data
- Duff, Tom, 233
- E**
 - edge flags, 63–64
 - tessellated polygons generate, 492
 - vertex arrays, specifying values with, 70
 - emission, 182, 208, 215
 - enabling
 - alpha test, 462
 - antialiasing of points or lines, 243
 - antialiasing polygons, 254
 - blending, 229
 - color material properties mode, 211
 - color sum mode, 444
 - culling, 58
 - depth test, 470
 - dithering, 167, 471
 - evaluators, 521, 525
 - fog, 255
 - lighting, 205
 - line stippling, 53
 - logical operations, 472
 - multisampling, 250
 - normal vectors for evaluated surfaces,
 - automatic generation of, 525, 533
 - polygon offset, 268
 - polygon stippling, 59
 - rescaling normals, 66, 186
 - stencil test, 464
 - texture coordinate generation, 427
 - texturing, 365, 369
 - unit length normal vectors ensured, 66, 186
 - endianness, 321
 - environment mapping, 428, 430
 - errata, xxiii
 - error handling, 583–585
 - error string description, 585
 - recommended usage, 704
 - evaluators, 518–532
 - basis functions, 517, 521
 - evenly spaced values, 523, 527
 - one-dimensional, 518
 - rendering pipeline stage, 11
 - sample program using mesh for 2D Bézier surface, 528
 - sample program which draws 1D Bézier curve, 519
 - sample program which draws 2D Bézier surface, 526
 - sample program which generates texture coordinates, 530
 - tessellation usage, 706
 - texture coordinates, generating, 530
 - two-dimensional, 526
 - event management, using GLUT, 19
 - example programs, *See* programs
 - extensions
 - Architecture Review Board approved, 587
 - Microsoft Windows and
 - wglGetProcAddress(), 589
 - vendor-specific, 587
 - eye coordinates, 102, 145
 - texture coordinate generation, 423, 427





F

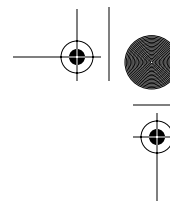
fade effect, 590
Farin, Gerald E., 517
feedback, 573–580
 array contents, 579
 pass-through markers, 576
 querying current rendering mode, 554
 returned data, 575
 sample program, 577
 steps to perform, 574
 tessellation, obtaining vertex data after, 503
Feiner, Steven K., xxi, 697
field of view, 106
 calculate, using trigonometry to, 138
filtering, 400–402
 mipmapped textures, 390–400, 402
 texture border colors, 418
flat shading, 173
flight simulation
 fog, use of, 255
flushing, 34, 704
fog, 255–265
 blending factors, 258
 color-index mode, 260
 density, 259
 enabling, 255
 equations, 258
 fog coordinates, 262
 hints, 255
 RGBA mode, 259
 sample program in color-index mode, 260
 sample program in RGBA mode, 256
 sample program with fog coordinates, 263
Foley, James D., xxi, 161, 165, 697
fonts, 303–306
 antialiased characters (by masking), 595
 antialiased characters (by texturing), 606
 bitmapped, 304
 creating with display lists, 287
 drawing, 302
 drawing as bitmaps, 298
 multi-byte, 304
 same program, 305

 sample program using multiple display lists, 288
 X fonts, using, 665
Foran, Jim, 445
foreshortening, perspective, 128
fragments, 452
 alpha test, 462
 blending, 225
 depth test, 469
 rendering pipeline operations, 13, 617
 scissor test, 462
 tests, 461–472
 texture functions, 412
framebuffer, 164, 453
 capacity per pixel, 454
 clearing, 456–457
 copying pixel data within, 307, 315, 316
 enabling for reading, 458
 enabling for writing, 458
 minimum configuration with the X Window System, 454
 querying color resolution, 164
 reading pixel data from, 307, 309
 writing pixel data to, 307, 313
front-facing polygons, 57
 specifying material property for, 206
 two-sided lighting, 203
frustum, 128
ftp (file-transfer protocol) site
 GLX specification, 662
 OpenGL Programming Guide, xxii
Fundamentals of Computer Aided Geometric Design, 517

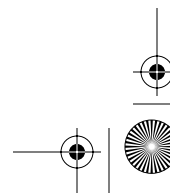
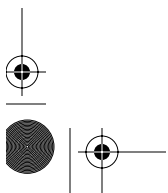
G

Game of Life, 609
gamma correction, 165
Gardner, Martin, 609
geometric primitives, 37–49, 614–616
 performance when specifying, 707
 rendering pipeline stage, 12
geosciences
 use of texturing in applications, 423



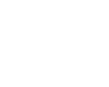


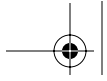
- giraffe, 168
- glAccum(), 474
- glActiveTextureARB*(), 434
- glAlphaFunc(), 463
- glAreTexturesResident(), 408
- glArrayElement(), 73
 - legal between glBegin() and glEnd(), 48
- Glassner, Andrew S., xxi
- glBegin(), 43, 492
 - restrictions, 47
- glBindTexture(), 369, 405
 - multitexturing, 434
- glBitmap(), 297, 301
 - feedback mode, 575
 - fonts, used for, 304
 - imaging pipeline operations, 317
 - pixel-storage modes effect, 320
- glBlendColor*(), 229
- glBlendFunc(), 227
- glBlendFuncSeparate(), 227
- glCallList(), 276, 279, 284
 - legal between glBegin() and glEnd(), 48
- glCallLists(), 286
 - fonts, use for, 303
 - legal between glBegin() and glEnd(), 48
 - sample program, 305
- glClear(), 31, 457, 618
 - depth buffer, clearing the, 180
- glClearAccum(), 32, 457
- glClearColor(), 30, 31, 457
- glClearDepth(), 31, 457
- glClearIndex(), 32, 173, 457
 - fog, use with, 260
- glClearStencil(), 32, 457
- glClientActiveTextureARB*(), 437
- glClipPlane(), 145
- glColor*(), 33, 171
 - legal between glBegin() and glEnd(), 47
- glColorMask(), 457, 459
- glColorMaterial(), 211
 - performance tips, 706
- glColorPointer(), 70
- glColorSubTable(), 342
- glColorTable(), 338
- glColorTableParameter(), 340
- glCompressedTexImage1D(), 387
- glCompressedTexImage2D(), 387
- glCompressedTexImage3D(), 387
- glCompressedTexSubImage1D(), 388
- glCompressedTexSubImage2D(), 388
- glCompressedTexSubImage3D(), 388
- glConvolutionFilter1D(), 349
- glConvolutionFilter2D(), 344
- glConvolutionParameter*(), 350
- glCopyColorSubTable(), 342
- glCopyColorTable(), 341
- glCopyConvolutionFilter1D(), 349
- glCopyConvolutionFilter2D(), 347
- glCopyPixels(), 307, 315
 - alternative uses, 611
 - dithering, turn off, 704
 - feedback mode, 575
 - glReadBuffer() effect, 459
 - imaging pipeline operations, 316
 - pixel-transfer modes effect, 324
- glCopyTexImage1D(), 380
 - glReadBuffer() effect, 459
 - pixel-transfer modes effect, 324
- glCopyTexImage2D(), 373
 - glReadBuffer() effect, 459
 - pixel-transfer modes effect, 324
- glCopyTexSubImage1D(), 381
 - glReadBuffer() effect, 459
 - pixel-transfer modes effect, 324
- glCopyTexSubImage2D(), 378
 - glReadBuffer() effect, 459
 - pixel-transfer modes effect, 324
- glCopyTexSubImage3D(), 384
 - pixel-transfer modes effect, 324
- glCullFace(), 58
- glDeleteLists(), 286, 304
- glDeleteTextures(), 407





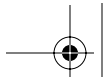
- glDepthFunc(), 470
- glDepthMask(), 459
 - blending opaque and translucent objects, 238
- glDepthRange(), 135
 - gluUnProject(), relationship to, 155
 - glWindowPos() effect, 300
- glDisable(), 10, 50
- glDisableClientState(), 69
- glDrawArrays(), 78
- glDrawBuffer(), 314, 315, 458
- glDrawElements(), 74
- glDrawPixels(), 307, 313, 465, 607
 - alternative uses, 611
 - feedback mode, 575
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glDrawRangeElements(), 77
 - version, 29
- glEdgeFlag(), 64
 - legal between glBegin() and glEnd(), 48
- glEdgeFlagPointer(), 70
- glEnable(), 50, 187
 - also see* enabling
- glEnableClientState(), 48, 68
- glEnd(), 43, 44, 492
 - restrictions, 47
- glEndList(), 275, 279, 281
- glEvalCoord*(), 523, 525
 - legal between glBegin() and glEnd(), 48
 - used instead of glVertex*(), 518, 521
- glEvalMesh*(), 524, 527
- glEvalPoint*()
 - legal between glBegin() and glEnd(), 48
- glFeedbackBuffer(), 574
 - glRenderMode(), use with, 554
- glFinish(), 35
- glFlush(), 35, 704
- glFog*(), 259
- glFogCoord*(), 263, 266
- glFrontFace(), 58
- glFrustum(), 107, 128, 129, 618
- glGenLists(), 275, 280
 - fonts, use for, 303
- glGenTextures(), 369, 404
- glGetBooleanv(), 10, 50, 621
 - double-buffering support, querying, 455
 - stereo support, querying, 455
- glGetClipPlane(), 620
- glGetColorTable(), 620
 - pixel-storage modes effect, 320
- glGetColorTableParameter*(), 620
- glGetCompressedTexImage(), 620
- glGetConvolutionFilter(), 620
 - pixel-storage modes effect, 320
- glGetConvolutionParameter*(), 620
- glGetDoublev(), 10, 50, 621
- glGetError(), 10, 584, 620
- glGetFloatv(), 10, 50, 621
 - line width attributes, obtaining, 52
- glGetHistogram(), 354, 620
 - pixel-storage modes effect, 320
- glGetHistogramParameter*(), 620
- glGetIntegerv(), 10, 50, 621
 - alpha test information, obtaining, 463
 - attribute stack depth, obtaining, 84
 - clipping planes, obtaining number of additional, 145
 - color resolution, obtaining, 164
 - display list nesting limit, obtaining, 284
 - matrix stack depth, obtaining, 143
 - maximum texture size, obtaining, 374
 - name stack depth, obtaining, 555
 - pixel map information, obtaining, 327
 - rendering mode, obtaining current, 554
 - stencil-related values, obtaining, 465
 - vertex array range values, obtaining, 78
- glGetLight*(), 10, 620
- glGetMap*(), 620
- glGetMaterial*(), 620
- glGetMinmax(), 357, 620
 - pixel-storage modes effect, 320
- glGetMinmaxParameter*(), 620
- glGetPixelMap*(), 621
- glGetPointerv(), 10, 50, 621



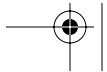


- glGetPolygonStipple(), 10, 621
- glGetSeparableFilter(), 621
 - pixel-storage modes effect, 320
- glGetString, 588
- glGetString(), 585, 621
- glGetTexEnv*(), 621
- glGetTexGen*(), 621
- glGetTexImage(), 621
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glGetTexLevelParameter*(), 375, 621
- glGetTexParameter*(), 621
 - texture residency, obtaining, 408
- glHint(), 242
 - fog use, 255
 - texture use, 369
- glHistogram(), 353
- glIndex*(), 172
 - fog, use with, 260
 - legal between glBegin() and glEnd(), 47
- glIndexMask(), 457, 459
- glIndexPointer(), 70
- glInitNames(), 553, 554, 555
- glInterleavedArrays(), 81
- glIsEnabled(), 10, 50, 621
- glIsList(), 286, 621
- glIsTexture(), 404, 621
- glLight*(), 187, 188, 189, 194
- glLightModel*(), 202
- glLineStipple(), 53
- glLineWidth(), 52
- glListBase(), 286
 - fonts, use for, 303
 - sample program, 305
- glLoadIdentity(), 107, 109, 119, 618
 - performance tips, 706
 - viewing transformations, use before, 105
- glLoadMatrix*(), 108, 109, 110, 618
- glLoadName(), 554, 556
- glLoadTransposeMatrix*(), 108, 110
- glLogicOp(), 472
- glMap*(), 520, 522, 525
- glMapGrid*(), 523, 527
- glMaterial*(), 188, 206
 - legal between glBegin() and glEnd(), 47
 - performance tips, 706
- glMatrixMode(), 107, 109
 - use with matrix stacks, 141
- glMinmax(), 356
- glMultiDrawArrays(), 79
 - version, 29
- glMultiDrawElements(), 76
 - version, 29
- glMultiTexCoord*ARB(), 435
- glMultMatrix*(), 108, 109, 618
 - performance tips, 706
- glMultTransposeMatrix*(), 108, 110
- glNewList(), 275, 279, 281
- glNormal*(), 65
 - legal between glBegin() and glEnd(), 47
- glNormalPointer(), 70
- glOrtho(), 132, 618
 - picking matrix use, 561
- glPassThrough(), 574, 576
- glPixelMap*(), 327
- glPixelStore*(), 320, 384
 - cannot be stored in display lists, 283
 - polygon stippling, 59
 - texture image data, effect on, 371, 373, 376, 378, 380, 383, 387, 388
- glPixelTransfer*(), 324, 607
 - texture image data, effect on, 371, 373, 376, 378, 380, 383, 387, 388
- glPixelZoom(), 328, 593
- glPointSize(), 51
- glPolygonMode(), 57
 - antialiasing, effect on, 254
 - polygon offset, use with, 268
- glPolygonOffset(), 268
- glPolygonStipple(), 59
 - pixel-storage modes effect, 320
- glPopAttrib(), 10, 84, 292, 435, 622
- glPopClientAttrib(), 10, 86, 435, 622





- glPopMatrix(), 141, 152, 197, 292
 - restore orientation of coordinate systems, 154
 - selection, use with, 553
- glPopName(), 554, 555
- glPrioritizeTextures(), 409
- glPushAttrib(), 10, 84, 292, 435, 622
- glPushClientAttrib(), 10, 86, 435, 622
- glPushMatrix(), 141, 152, 197, 292
 - save orientation of coordinate systems, 154
 - selection, use with, 553
- glPushName(), 553, 554, 555
- glRasterPos*(), 297, 299
 - images, for positioning, 307
 - multitexturing, with, 436
 - selection hit, 556
- glReadBuffer(), 314, 315, 459
- glReadPixels(), 307, 309
 - glReadBuffer() effect, 459
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glRect*(), 41
- glRenderMode(), 553, 554, 556, 574
- glResetHistogram(), 356
- glResetMinmax(), 358
- glRotate*(), 116, 148, 151, 618
 - performance tips, 706
- glSampleCoverage(), 253
- glScale*(), 105, 116, 151, 618
 - performance tips, 706
- glScissor(), 462
- glSecondaryColor*(), 444
- glSelectBuffer(), 553, 554
 - display lists, cannot be stored in, 283
- glSeparableFilter2D(), 348
- glShadeModel(), 173
- glStencilFunc(), 464
- glStencilMask(), 459
- glStencilOp(), 464
- glTexCoord*(), 369, 415
 - legal between glBegin() and glEnd(), 48
 - texture unit 0, for, 436
- glTexCoordPointer(), 70
- glTexEnv*(), 369, 410, 438
 - level of detail bias, 394
 - multitexturing, 433
- glTexGen*(), 422
 - cube maps, 431
 - environment mapping, 429
 - multitexturing, 433, 436
- glTexImage1D(), 379
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glTexImage2D(), 369, 370
 - cube map textures, 430
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
 - specifying mipmaps, 391
- glTexImage3D(), 382
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glTexParameter*(), 369, 420
 - automatic mipmap regeneration, 400
 - mipmap level of detail, controlling, 397
 - mipmap levels, controlling base and maximum, 396
 - multitexturing, 433
 - specifying filtering methods, 401
- glTexSubImage1D(), 380
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glTexSubImage2D(), 376
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glTexSubImage3D(), 383
 - pixel-storage modes effect, 320
 - pixel-transfer modes effect, 324
- glTranslate*(), 115, 148, 151, 618
 - performance tips, 706
- GLU, 3, 14, 488
 - drawing spheres, cylinders, and disks, 505–514
 - error string description, 585
 - obsolete routines
 - gluBeginPolygon(), 504
 - gluEndPolygon(), 504
 - gluNextContour(), 504





quadrics, 505–514
tessellation, 40, 488–505
version numbers, obtaining, 586
`gluBeginCurve()`, 533, 543
`gluBeginSurface()`, 533, 541
`gluBeginTrim()`, 547
`gluBuild1DMipmapLevels()`, 399
`gluBuild1DMipmaps()`, 399
`gluBuild2DMipmapLevels()`, 399
`gluBuild2DMipmaps()`, 399
`gluBuild3DMipmapLevels()`, 399
`gluBuild3DMipmaps()`, 399
`gluCheckExtension()`, 588
`gluCylinder()`, 506, 509
`gluDeleteNurbsRenderer()`, 537
`gluDeleteQuadric()`, 506, 507
`gluDeleteTess()`, 503, 504
`gluDisk()`, 506, 510
`gluEndCurve()`, 533, 543
`gluEndSurface()`, 533, 541
`gluEndTrim()`, 547
`gluErrorString()`, 507, 541, 585
 polygon tessellation, 492
`gluGetNurbsProperty()`, 540, 621
`gluGetString`, 588
`gluGetString()`, 587, 621
`gluGetTessProperty()`, 499, 621
`gluLoadSamplingMatrices()`, 540
`gluLookAt()`, 103, 105, 123, 148
`gluNewNurbsRenderer()`, 533, 537
`gluNewQuadric()`, 506, 507
`gluNewTess()`, 490, 504
`gluNurbsCallback()`, 533, 541, 544
`gluNurbsCallbackData()`, 545
`gluNurbsCurve()`, 533, 543
`gluNurbsProperty()`, 533, 538
 returning tessellated data, 543
`gluNurbsSurface()`, 533, 542
`gluOrtho2D()`, 132, 705
 resized windows, use with, 36
`gluPartialDisk()`, 506, 510
`gluPerspective()`, 107, 130, 148
 picking matrix use, 561
`gluPickMatrix()`, 561
`gluProject()`, 158
`gluPwlCurve()`, 547
`gluQuadricCallback()`, 506, 507
`gluQuadricDrawStyle()`, 506, 507
`gluQuadricNormals()`, 506, 508
`gluQuadricOrientation()`, 506, 508
`gluQuadricTexture()`, 506, 508
`gluScaleImage()`, 372
`gluSphere()`, 506, 509
GLUT, 15, 683–689
 basic functions, 16–20
 event management, 19
 `glutCreateWindow()`, 17, 685
 `glutDisplayFunc()`, 18, 685
 `glutIdleFunc()`, 20, 689
 `glutInit()`, 17, 684
 `glutInitDisplayMode()`, 17, 684
 `glutInitWindowPosition()`, 17, 685
 `glutInitWindowSize()`, 17, 685
 `glutKeyboardFunc()`, 19, 686
 `glutMainLoop()`, 18, 689
 `glutMotionFunc()`, 20, 686
 `glutMouseFunc()`, 19, 686
 `glutPostRedisplay()`, 18, 276, 687
 `glutReshapeFunc()`, 19, 686
 simple example, 36
 `glutSetColor()`, 17, 173, 221, 687
 smooth shading, use for, 175
 `glutSolidCone()`, 688
 `glutSolidCube()`, 20, 688
 `glutSolidDodecahedron()`, 688
 `glutSolidIcosahedron()`, 688
 `glutSolidOctahedron()`, 688
 `glutSolidSphere()`, 20, 687
 `glutSolidTeapot()`, 688
 `glutSolidTetrahedron()`, 688
 `glutSolidTorus()`, 688
 `glutSwapBuffers()`, 24
 `glutWireCone()`, 688
 `glutWireCube()`, 20, 688
 `glutWireDodecahedron()`, 688
 `glutWireIcosahedron()`, 688





GLUT (*Continued*)

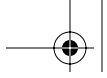
- glutWireOctahedron(), 688
- glutWireSphere(), 20, 148, 687
- glutWireTeapot(), 688
- glutWireTetrahedron(), 688
- glutWireTorus(), 688
- multisampling, 250
- sample program introducing GLUT, 18
- window management, 17, 36
- gluTessBeginContour(), 501
- gluTessBeginPolygon(), 500
- gluTessCallback(), 491, 501, 504
- gluTessEndContour(), 501
- gluTessEndPolygon(), 500
- gluTessNormal(), 499, 500, 503
- gluTessProperty(), 496, 501
- gluTessVertex(), 501, 504
- gluUnProject(), 155, 158
- gluUnProject4(), 158
- glVertex*(), 42
 - legal between glBegin() and glEnd(), 47
 - using glEvalCoord*() instead, 518
- glVertexPointer(), 48, 70
- glViewport(), 108, 134
 - using with resized windows, 36
- glWindowPos*(), 300
 - multitexturing, with, 436
 - selection hit, 556
- GLX, 14, 662
 - ftp site for GLX specification, 662
 - glXChooseFBConfig(), 662, 665
 - glXChooseVisual(), 662, 667, 708
 - glXCopyContext(), 664, 666
 - glXCreateContext(), 664, 668
 - glXCreateGLXPixmap(), 663, 668
 - glXCreateNewContext(), 663, 666
 - glXCreatePbuffer(), 662, 666
 - glXCreatePixmap(), 662, 666
 - glXCreateWindow(), 662, 666
 - glXDestroyContext(), 664, 667
 - glXDestroyGLXPixmap(), 668
 - glXDestroyPbuffer(), 665, 667
 - glXDestroyPixmap(), 665, 667
 - glXDestroyWindow(), 665, 667

- glXGetClientString(), 662, 665
- glXGetConfig(), 454, 663, 667
- glXGetCurrentContext(), 663, 666
- glXGetCurrentDisplay(), 663, 666
- glXGetCurrentDrawable(), 663, 666
- glXGetCurrentReadDrawable(), 666
- glXGetFBConfigAttrib(), 662, 666
- glXGetFBConfigs(), 665
- glXGetSelectedEvent(), 664, 667
- glXGetVisualFromFBConfig(), 662, 666
- glXIsDirect(), 663, 666
- glXMakeContextCurrent(), 663, 666
- glXMakeCurrent(), 664, 668
- glXQueryContext(), 663, 667
- glXQueryExtension(), 662, 665
- glXQueryExtensionsString(), 662, 665
- glXQueryServerString(), 662, 665
- glXQueryVersion(), 662, 665
- glXSelectEvent(), 664, 667
- glXSwapBuffers(), 23, 664, 667
- glXUseXFont(), 665, 667
- glXWaitGL(), 664, 667
 - performance tips, 708
- glXWaitX(), 664, 667
 - performance tips, 708
- glXGetCurrentReadDrawable(), 663, 664
- glXQueryExtensionString(), 587
- Gouraud shading, *See* smooth shading

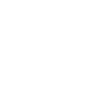
H

- Haeberli, Paul, 445, 473
- haze, *See* fog
- header file, 15
- hidden-line removal, 604
 - polygon offset used for, 268
- hidden-surface removal, 179–181, 469
- hierarchical models, 140, 284
 - picking, 565–567
- highlights, *See* specular
- hints, 242
 - fog, 255
 - perspective correction, 242, 369





- histogram, 353–356
 - resetting, 354, 356
 - retrieving, 354
 - sample program, 354
- hits (selection), *See* selection (hit records)
- holes in polygons, 39, 601
- homogeneous coordinates, 38, 698
- Hoschek, Josef, 517
- Hughes, John F., xxi, 697
- I**
- IBM OS/2 Presentation Manager to OpenGL Interface, *See* PGL
- icosahedron, drawing, 88
- identity matrix, 105, 109, 119, 706
- illumination, *See* lighting
- images, 296, 306–315
 - also see* pixel data
 - blending, 598
 - compositing, 226
 - distorted, 593
 - imaging pipeline, 315–331
 - interpolating between, 598
 - magnifying or reducing, 328
 - nonrectangular, 233
 - projecting, 606
 - sample code which draws an image, 314
 - sample program which draws, copies, and zooms an image, 329
 - scaling and rotating, 606
 - sources of, 307
 - superimposing, 599
 - transposing, 611
 - warping, 606
- imaging pipeline, *See* images (imaging pipeline)
- imaging subset, 336–358
 - extension string, 588
 - texture images, effect on, 372, 379
- immediate mode, 29, 272
 - display lists, mixing with, 283
- infinite light source, 191
- input events
 - handling, using GLUT, 19
- intensity
 - texture image data type, 411
- Interactive Inspection of Solids: Cross-sections and Interferences*, 601
- interference regions, 601
- interleaved arrays, 80
- interpolating
 - color values and texture coordinates, 242, 414
 - texture combiner function, 442
- invariance
 - of an OpenGL implementation, 704, 709
- J**
- jaggies, 241
- jittering, 474, 479, 486
 - accFrustum() routine, 475
 - accPerspective() routine, 475
 - sample code to jitter projection transformations, 475
 - sample program with orthographic projection, 479
- K**
- Kilgard, Mark, xxii, 15, 662, 683
- Korobkin, Carl, 445
- L**
- Lasser, Dieter, 517
- layers, drawing, 594
- Life, Game of, 609
- light sources, 188–201
 - ambient light, 182, 190
 - contribution to lighting equation, 216
 - diffuse light, 182, 190
 - directional, 191





light sources (*Continued*)

- display lists cache values, 277
- infinite light source, 191
- local light source, 191
- maximum number of sources, 187
- moving along with the viewpoint, 199
- moving light sources, 195–200
- multiple light sources, 194
- performance tips, 187
- positional, 191
- rendering pipeline stage, 12, 615
- RGBA values, 183
- sample program that moves the light source, 198
- specifying a light source, 187
- specular light, 182
- spotlights, 193–194
- stationary, 196

lighting

- also see* light sources, material properties
- ambient light, 181
- approximation of the real world, 181
- attenuation, 191–192
- calculations in color-index mode, 221
- color-index mode, 220–222
- default values, using, 188
- display lists cache values, 277
- enabling, 187, 188
- enabling and disabling, 205
- equation that calculates lighting, 215
- global ambient light, 202, 216
- lighting model, 201–204
- lighting model, specifying a, 187
- rendering pipeline stage, 12, 615
- sample program introducing lighting, 184
- specular color separated, 204, 219, 444
- steps to perform, 184
- two-sided materials, 203
- viewer, local or infinite, 203

line segment, 38

linear attenuation, 192

lines, 38

- antialiasing, 243–249, 606
- connected closed loop, specifying, 44, 46
- connected strip, specifying, 44, 46
- feedback mode, 575

- querying line width, 52

- sample program with wide, stippled lines, 55

- specifying, 44, 46

- stippling, 53

- tessellated polygons decomposed into, 492
- width, 52

- local light source, 191

logical operations

- rendering pipeline stage, 14, 617

- transposing images, using for, 611

- lookup table, *See* color map

- luminance, 309, 335

- pixel data formats for, 311, 319

- texture image data type, 411

M

- magnifying images, 328

- masking, 459

- antialiasing characters, 597

- layers, drawing, 594

- rendering pipeline stage, 14, 617

- material properties, 188, 205–214

- ambient, 183, 207

- changing a single parameter with `glColorMaterial()`, 211

- changing material properties, 209

- diffuse, 183, 207

- display lists cache values, 277

- emission, 182, 208, 215

- enabling color material properties mode, 211

- performance when changing, 706

- rendering pipeline stage, 12, 615

- RGBA values, 184

- sample program which changes material properties, 209

- sample program which uses `glColorMaterial()`, 212

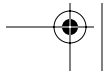
- shininess, 208

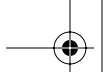
- specular, 183, 208

- two-sided lighting, 203

- matrix

- also see* matrix stack



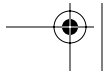


- choosing which matrix is current, 109
 - column-major ordering, 110
 - current, 105
 - danger of extensive changes, 704
 - display lists cache matrix operations, 277
 - identity, 105, 109, 119, 706
 - loading, 109
 - loading transposed, 110
 - modelview, 102, 109
 - multiplying matrices, 109
 - multiplying transposed matrices, 110
 - NURBS, specifying for sampling, 539
 - orthographic parallel projection, 702
 - perspective projection, 701
 - projection, 107, 109
 - rotation, 700
 - row-major ordering, 110
 - scaling, 700
 - texture, 445
 - transformation pipeline, 100
 - transformations of homogeneous coordinates, 698
 - translation, 700
 - matrix stack, 139–144
 - choosing which matrix stack is current, 141
 - current matrix stack, 618
 - modelview, 143
 - popping, 141
 - projection, 143
 - pushing, 141
 - querying stack depth, 143
 - texture, 445
 - Megahed, Abe, 601
 - Microsoft
 - callback functions on Windows, 493
 - Microsoft Win32, *See* Win32
 - Microsoft Windows 95/98/NT, xxii, 14, 677
 - Microsoft Windows to OpenGL interface, *See* WGL
 - minmax, 356–358
 - resetting, 357, 358
 - retrieving results, 357
 - sample program, 357
 - mipmapping, 390–400
 - automated generation, 398
 - base and maximum levels, 396
 - level of detail control, 395
 - minification filters, 402
 - texture objects for mipmaps, 407
 - mirroring objects, *See* scaling
 - modeling transformations, 105, 111, 114–119
 - camera analogy, 100
 - connection to viewing transformations, 105
 - example, 117
 - rotation, 116
 - rotation matrix, 700
 - sample program, 119
 - scaling, 116
 - scaling matrix, 700
 - translation, 115
 - translation matrix, 700
 - models
 - rendering wireframe and solid, 20, 687
 - modelview matrix, 102, 109
 - arbitrary clipping planes, effect on, 145
 - stack, 143
 - mosaicing, 395
 - motion blur, 480
 - stippling, with, 591
 - motion, *See* animation
 - movie clips, 611
 - multiple layers
 - displaying with overlap, 594
 - multisampling, 249–253
 - fading point primitives, 266
 - sample program, 250
 - multitexture
 - extension string, 588
 - multitexturing, 432–437
- ## N
- name stack, 553–557
 - creating, 554
 - initializing, 554
 - loading, 554





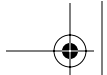
- name stack (*Continued*)
 - multiple names, 565–567
 - popping, 554
 - pushing, 554
 - querying maximum depth, 555
 - networked operation, 34–35
 - attribute groups, saving and restoring, 84
 - display lists, 283
 - versions, 586
 - Non-Uniform Rational B-Splines, *See* NURBS
 - nonplanar polygons, 40
 - normal vectors, 64–66, 186
 - calculating, 692
 - calculating for analytic surfaces, 693
 - calculating for polygonal data, 695
 - calculating length, 66
 - cross product, calculating normalized, 90
 - enabling automatic unit length division, 66, 186
 - inverse matrix generated, 618
 - matrix transformations, 102
 - normalized, 66
 - NURBS, generating for, 542
 - quadrics, generated for, 508
 - rendering pipeline stage, 12, 615
 - specifying, 65
 - tessellation, specifying for, 494
 - transformations, 699
 - uniform rescaling, 66
 - unit length optimizes performance, 707
 - vertex arrays, specifying values with, 70
 - normal, *See* normal vectors
 - normalized device coordinates, 103
 - NURB Curves and Surfaces*, 517
 - NURBS, 532–550
 - creating a NURBS curve or surface, 541–543
 - creating a NURBS object, 537
 - culling, 538
 - deleting a NURBS object, 537
 - display list use, 276
 - error handling, 540
 - method of display (lines or filled polygons), 538
 - normal vectors, generating, 542
 - properties, controlling NURBS, 537
 - querying property value, 540
 - references, 517
 - sample program which draws a lit NURBS surface, 534
 - sample program with a trimmed surface, 549
 - sampling precision, 538
 - source for matrices, 539
 - steps to use, 533
 - texture coordinate generation, 542
 - trimming, 547–550
 - NURBS Book, The*, 517
 - NURBS for Curve and Surface Design*, 517
 - NURBS tessellator
 - sample code, 545, 546
- O**
- object coordinates, 102
 - texture coordinate generation, 423
 - objects, *See* models
 - opacity, 226
 - OpenGL Extension to the X Window System,
see GLX
 - OpenGL Programming for the X Window System*,
xxii, 15, 17, 662, 683
 - OpenGL Reference Manual*, xxi, 613, 620, 662
 - OpenGL Utility Library, *See* GLU
 - OpenGL Utility Toolkit, *See* GLUT
 - orthographic parallel projection, 107,
131–132
 - jittering, 479
 - matrix, 702
 - specifying with integer coordinates, 705
 - outer product, 347
 - outlined polygons, 57, 64
 - polygon offset solution, 268
 - overlapping objects, 601
- P**
- packed pixel data, 311–312
 - painting, 226, 232, 611





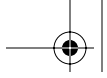
- partial disks, 506
- pass-through markers, 576
- performance tips
 - clearing the window, 32
 - display lists, 276
 - flat shading, 707
 - flushing the pipeline, 34
 - fog, 255
 - GLX tips, 708
 - hints, 242
 - light source attenuation, effect of, 192
 - light sources, effect of additional, 187
 - list of general tips, 706
 - material properties, changing, 706
 - NURBS and display lists, 276
 - pixel data alignment, 322
 - pixel data, drawing, 335
 - polygon restrictions, 39
 - polygon subdivision, 87
 - pushing and popping attribute groups, 706
 - rasterization and fragment operations for pixel data, 707
 - removing hidden surfaces, 181
 - specifying geometric primitives, 707
 - tessellation and display lists, 276
 - tessellation, use of, 503
 - texture images, internal format of, 371
 - texture objects, 403, 706
 - texture subimages, 706
 - two-sided lighting, 204
 - unit-length normal vectors, 707
 - vector and scalar forms of commands, 707
 - vertex arrays, 707
- perspective projection, 128–131
 - correction hint, 242, 369
 - depth coordinates, effect on, 136
 - jittering, 475
 - matrix, 701
 - perspective division, 102
- PGL, 14, 674
 - pglChooseConfig(), 674, 676
 - pglCopyContext(), 675, 676
 - pglCreateContext(), 675, 676
 - pglDestroyContext(), 675, 676
 - pglGetCurrentContext(), 675, 676
 - pglGetCurrentWindow(), 675, 676
 - pglGrabFrontBitmap(), 675, 677
 - pglIsIndirect(), 675, 676
 - pglMakeCurrent(), 675, 676
 - pglQueryCapability(), 674, 676
 - pglQueryConfigs(), 674, 676
 - pglQueryVersion(), 674, 676
 - pglReleaseFrontBitmap(), 675, 677
 - pglSelectColorIndexPalette(), 675, 677
 - pglSwapBuffers(), 675, 677
 - pglUseFont(), 676, 677
 - pglWaitGL(), 675, 677
 - pglWaitPM(), 675, 677
- picking, 560–570
 - back buffer for, using the, 592
 - depth coordinates, 567
 - hierarchical models, 565–567
 - projection matrix, special, 561
 - sample program, 562
 - sample program with depth coordinates, 567
 - strategies, 571
 - sweep selection, 572
- Piegl, Les, 517
- pipeline
 - geometric processing, 615–616
 - imaging, 315–331
 - rendering, 10–14
 - vertex transformation, 100
- pixel
 - coverage, 241
- pixel data, 296, 306–315
 - also see* images
 - BGR and BGRA formats, 309
 - byte alignment, 322
 - byte swapping, 321
 - copying within the framebuffer, 13, 307, 315, 316, 617
 - depth buffer pixel data, 315, 326
 - drawing or reading a subrectangle of, 322
 - drawing process in detail, 332–333
 - endianness, 321
 - feedback mode, 575
 - formats for reading or drawing, 309
 - formats for storing in memory, 311, 319
 - mapping, 13, 327–328, 616





- pixel data (*Continued*)
 - packed, 311–312
 - packing into processor memory, 13, 318–321, 616
 - performance tips, 335
 - pipeline operations, 12, 315–331, 616
 - pixel zoom, 328
 - querying pixel mapping information, 327
 - reading from the framebuffer, 307, 309
 - reading process in detail, 334–335
 - sample code which draws an image, 314
 - sample program which draws, copies, and zooms pixel data, 329
 - stencil buffer pixel data, 311, 326
 - storage modes, 319, 384–386
 - transfer modes, 13, 324, 411, 616
 - unpacking from processor memory, 12, 318–321, 616
 - writing to the framebuffer, 307, 313
- point light source, *See* positional light source
- point parameters, 265–267
 - sample program, 267
- points, 38
 - antialiasing, 243–249, 598
 - drawing, 43
 - feedback mode, 575
 - point parameters, 265–267
 - round, 243–249, 598
 - size, 51
 - specifying, 44, 46
- polygon offset, 268–270
 - depth slope of a polygon, 269
 - enabling, 268
 - hidden-line removal, 604
 - shadowing use, 447
- polygonal approximations to surfaces, 86
- polygons, 38
 - boundary edges, 63–64
 - concave, drawing filled, 488, 600
 - convex, 39
 - culling the faces, 57
 - drawing, 43
 - drawing as points, lines, or filled, 57
 - feedback mode, 575
 - front and back faces, 57
 - holes in, 39
 - non-convex, 39, 63
 - nonplanar, 40
 - polygon mode, 12, 57, 616, 707
 - reversing the faces, 57
 - sample program with stippled polygons, 61
 - self-intersecting, 495
 - simple, 39
 - specifying, 44, 47
 - stippling, 59
 - tessellation, specifying for, 500
 - Voronoi, 608
- positional light source, 191
- primitives
 - geometric, 37–49
 - raster, 296
- priority of texture objects, 409
- Procedural Elements for Computer Graphics*, 506
- programs
 - aaindex.c, 246
 - aargb.c, 244
 - accanti.c, 479
 - accpersp.c, 475
 - alpha3D.c, 238
 - alpha.c, 235
 - bezcurve.c, 519
 - bezmesh.c, 528
 - bezsurf.c, 526
 - checker.c, 367
 - clip.c, 146
 - colormat.c, 212
 - colormatrix.c, 352
 - colortable.c, 340
 - combiner.c, 442
 - convolution.c, 345
 - cube.c, 104
 - cubemap.c, 431
 - dof.c, 483
 - double.c, 24
 - drawf.c, 298
 - feedback.c, 577
 - fog.c, 256
 - fogcoord.c, 263
 - fogindex.c, 260
 - font.c, 305
 - ftp site, xxii





- hello.c, 18
 - histogram.c, 354
 - image.c, 329
 - light.c, 184
 - lines.c, 55
 - list.c, 279
 - material.c, 209
 - minmax.c, 357
 - mipmap.c, 392
 - model.c, 119
 - movelight.c, 198
 - multisamp.c, 250
 - multitex.c, 434
 - mvarray.c, 77
 - pickdepth.c, 567
 - picksquare.c, 562
 - planet.c, 149
 - pointp.c, 267
 - polys.c, 61
 - quadratic.c, 511
 - robot.c, 152
 - select.c, 557
 - shadowmap.c, 447–448
 - smooth.c, 174
 - stencil.c, 466
 - stroke.c, 288
 - surface.c, 534
 - surfpoints.c, 545, 546
 - tess.c, 492, 494, 502
 - texbind.c, 405
 - texgen.c, 424
 - texsub.c, 377
 - texture3d.c, 382
 - texturesurf.c, 530
 - torus.c, using a display list, 273
 - trim.c, 549
 - unproject.c, 156
 - varray.c, 71
 - projecting images, 606
 - projection matrix, 107, 109
 - matrix stack, 143
 - orthographic parallel projection matrix, 702
 - perspective projection matrix, 701
 - shadows created with, 603
 - projection transformations, 106, 127–132
 - camera lens analogy, 100
 - collapsing geometry to a single plane, 704
 - jittering, 475, 479
 - orthographic parallel, 107, 131–132, 705
 - perspective, 128–131
 - picking, 561
 - texturing effects, 445
 - two-dimensional, 132
 - proxies
 - color table, *see* color table proxies, 343
 - proxy textures, 374
 - cube maps, 430
- ## Q
- q texture coordinates, 445
 - avoiding negative values, 705
 - quadratic attenuation, 192
 - quadrics, 505–514
 - creating an object, 507
 - destroying an object, 507
 - drawing as points, lines, and filled polygons, 507
 - error handling, 507
 - normal vectors, generating, 508
 - orientation, 508
 - quadratic equation, 506
 - sample program, 511
 - steps to use, 506
 - texture coordinates, generating, 508
 - quadrilateral
 - specifying, 44
 - strip, specifying, 44, 47
- ## R
- raster position, 299
 - after drawing a bitmap, 301
 - current, 299, 618
 - current raster color, 302
 - current, obtaining the, 300

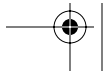


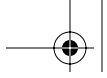


raster position (*Continued*)
 selection hit, 556
 transformation of, 299
rasterization, 164, 452
 exact, two-dimensional, 705
 rendering pipeline stage, 13
readImage(), 341
reading pixel data, *See* pixel data
Real Projective Plane, The, 697
rectangles
 specifying, 40
reducing images, 328
reflecting objects, *See* scaling
reflection, *See* material properties
reflective objects, *See* environment mapping
refresh, screen, 21
removing hidden surfaces, *See* hidden-surface removal
repeatability, 710
rescaling normals, 66, 186
resident textures, 375, 408
 management strategies, 409
 querying residence status, 408
RGBA mode, 165
 changing between color-index mode and, 170
 choosing between color-index mode and, 169
 coverage calculations for antialiasing, 241
 data type conversion, 171
 light source colors, 183
 lighting calculations in, 215
 material property values, 184
 vertex arrays, specifying values with, 70
Robins, Nate, xxiii, 120, 125, 132, 155, 201, 210, 403, 414, 415, 422, 446
robot arm example, 151–154
Rogers, David, 506
Rossignac, Jarek, 601
rotating images, 606
rotation, 116
 matrix, 700

S

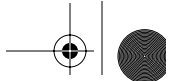
sample programs, *See* programs
scaling, 116
 matrix, 700
scaling images, 606
Schneider, Bengt-Olaf, 601
Scientific American, 609
scissor test, 462
 and clearing, 457
 rendering pipeline stage, 14, 617
secondary color, 443–444
 specular, 204, 219
Segal, Mark, 445
selection, 552–573
 back buffer for, using the, 592
 hit records, 556
 programming tips, 571
 querying current rendering mode, 554
 rendering pipeline stage, 616
 sample program, 557
 steps to perform, 553
 sweep selection, 572
shading
 flat, 173
 performance tips, 707
 sample program with smooth shading, 174
 smooth, 173
 specifying shading model, 173
shadows, 215, 485, 603
shininess, 208
 also see environment mapping
silhouette edges, 87
smoke, *See* fog
smooth shading, 173
solar system example, 148–151
source factor, *See* blending
specular
 contribution to lighting equation, 218
 light, 182
 material properties, 183, 208
 secondary specular color, 204, 219, 444





- sphere map, 428
 - spheres, 506, 687
 - split-screen
 - multiple viewports, 133
 - spotlights, *See* light sources
 - state machine, 9–10
 - state variables, 49
 - attribute groups, 83–86
 - display list execution, effect of, 292
 - enable and disable states, 50
 - list of, 621–657
 - performance of storing and restoring, 706
 - querying, 50
 - stencil buffer, 454, 455
 - clearing, 32, 457
 - concave polygons, for drawing, 600
 - decals, for, 599
 - Dirichlet domains, for, 609
 - Game of Life, for the, 609
 - hidden-line removal, 605
 - masking, 459
 - pixel data, 311, 326
 - stencil test, 464–469
 - examples of using, 465
 - interference regions found using clipping planes, 602
 - querying stencil parameters, 465
 - rendering pipeline stage, 14, 617
 - sample program, 466
 - stereo, 455, 458
 - querying its presence, 455
 - stippling
 - display lists cache stipple patterns, 277
 - enabling line stippling, 53
 - enabling polygon stippling, 59
 - fade effect, use for, 590
 - line pattern reset, 54, 575, 579
 - lines, 53
 - polygons, 59
 - sample program with line stipple, 55
 - sample program with polygon stippling, 61
 - stencil test, use of, 469
 - translucency, use to simulate, 590
 - stitching, 268
 - stretching objects, *See* scaling
 - stride
 - vertex arrays, 72, 81
 - subdivision, 86–94
 - generalized, 93
 - icosahedron example, 91
 - recursive, 93
 - subimages, 376–379, 380, 383
 - superimposing images, 599
 - surface normals, *See* normal vectors
 - surfaces, *See* evaluators or NURBS
 - swapping buffers, *See* double-buffering
 - syntax, *See* command syntax
- ## T
- Terminator 2*, 428
 - tessellation, 40, 488–505
 - backward compatibility with obsolete routines, 504
 - begin and end callback routines, 492
 - callback routines, 490–495
 - combine callback routine, 492, 495
 - contours, specifying, 500
 - converting code to use the GLU 1.2 tessellator, 505
 - creating an object, 490
 - decomposition into geometric primitives, 492
 - deleting objects, 503
 - display list use, 276
 - edge flag generation, 492
 - error handling, 492
 - evaluators used to perform, 706
 - interior and exterior, determining, 496–499
 - intersecting contours combined, 492, 495
 - performance tips, 503
 - polygons, specifying, 500
 - properties, 496–500
 - reuse of objects, 490, 504
 - reversing winding direction, 500
 - sample code, 492, 494, 502
 - user-specified data, 495





tessellation (*Continued*)

vertices, specifying, 493, 501
winding rules, 496–499

texels, 14, 361

text, *See* characters

texture coordinates, 369, 414–429
assigning manually, 414
avoiding negative q values, 705
clamping, 417–420
computing manually, 416
cube maps, 431
enabling automatic generation of, 427
environment mapping, automatic
generation for, 429
evaluators, generated by, 530
generating automatically, 422–429
multitexturing, special situations with,
436
NURBS, generating for, 542
q coordinate, 445
quadrics, generated for, 508
reference planes, specifying, 423
rendering pipeline stage, 12, 615
repeating, 417–420
sample program with texture coordinate
generation, 424
tessellation, specifying for, 494
vertex arrays, specifying values with, 70
wrapping modes, 417–420

texture functions, 410–414
add, 413
blend, 414
blending color, 414
decal, 369, 413
fragment operations, 412
level of detail bias, 410
modulate, 413
pixel-transfer modes effect, 411
replace, 413
texture internal format, interaction with,
412

texture images
alpha data, 411
borders, 389, 418
components, 370
compressed, 386

data types, 371
distorting, 416
framebuffer as a source of, 373, 378, 380,
384
imaging pipeline operations, 318
intensity data, 411
internal format, 370
luminance data, 411
mipmaps, 390–400
multitexturing, 433
one-dimensional, 379–381
performance affected by internal format,
371
performance of texture subimages, 706
power of 2 size restriction, 372
proxy textures, 374
querying maximum size, 374
residence status, 408
resident textures, 375, 408
resident textures, management strategies
of, 409
sample program with mipmaps, 392
sample program with subimages, 377
specifying, 369–389
subimages, 376–379, 380, 383
three-dimensional, 381–386
working set of textures, 375, 403, 408

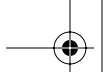
texture mapping
sample program using 3D textures, 382

texture mapping, *See* texturing

texture matrix, 445
rendering pipeline stage, 615

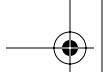
texture objects, 369, 403–408
binding, 404
creating, 404
data which can be stored in, 404
deleting, 407
fragmentation of texture memory, 410
least-recently used (LRU) strategy, 410
mipmaps, 407
naming, 404
performance tips, 403, 706
priority, 409
rendering pipeline, 13, 617
sample program, 367





- sample program with multiple texture objects, 405
- sharing among rendering contexts, 663, 678
- steps to perform, 403
- using, 404
- texturing
 - also see* texture coordinates, texture functions, texture images, texture matrix, and texture objects
 - antialiasing characters, 606
 - antialiasing lines, 606
 - blending, 233
 - border colors, treatment of, 418
 - color-index mode limitations, 365, 372
 - combiner functions, 437–443
 - compressed textures, 386
 - creating contours, 423
 - cube maps, 430
 - decals with alpha testing, 463
 - differences among releases, 363
 - enabling, 365, 369
 - filtering, 400–402
 - image transformations, 606
 - mipmapping, 390–400, 402
 - mosaic texture, 395
 - multitexturing, 432–437
 - perspective correction hint, 369
 - popping visual artifact, 395
 - rendering pipeline stage, 13, 617
 - sample code with a depth texture, 447–??
 - sample code with combiner functions, 442
 - sample code with multitexturing, 434
 - sample program, 367
 - sample program with cube maps, 431
 - sample program with evaluated, Bézier surface, 530
 - sample program with mipmapping, 392
 - sample program with texture coordinate generation, 424
 - sample uses for, 606
 - simulating shadows or spotlights, 445
 - specular color separated, 204, 219, 444
 - sphere map, 428
 - steps to perform, 365
- 3D Computer Graphics: A User's Guide for Artists and Designers, xxi
- 3D models, rendering, 20, 687
- Tiller, Wayne, 517
- tips, programming, 703
 - also see* performance tips
 - error handling, 704
 - selection and picking, 571
 - transformations, 136
- transformations
 - also see* modeling transformations, projection transformations, viewing transformations, and viewport transformations
 - combining multiple, 147–154
 - display lists cache transformations, 277
 - general-purpose commands, 108
 - matrices, 699–702
 - modeling, 111, 114–119
 - ordering correctly, 111–114
 - overview, 98
 - performance tips, 706
 - projection, 106, 127–132
 - reversing the geometric processing pipeline, 155
 - sample program, 104
 - sample program combining modeling transformations, 149, 152
 - sample program for modeling transformations, 119
 - sample program showing reversal of transformation pipeline, 156
 - troubleshooting, 136–139
 - units, 131
 - viewing, 111, 120–125
 - viewport, 108, 133–135
- translation, 115
 - matrix, 700
- translucent objects, 226, 590
 - stencil test, creating with the, 469
- transparent objects, 226
 - creating with the alpha test, 463
- transposing images, 611
- triangle
 - fan, specifying, 44
 - specifying, 44, 46
 - strip, specifying, 44, 46
 - tessellated polygons decomposed into, 492





trimming
 curves and curved surfaces, 547–550
 sample program, 549
tutorials
 on-line, xxiii
two-sided lighting, 203

U

up-vector, 105
Utility Library, OpenGL, *See* GLU
Utility Toolkit, OpenGL, *See* GLUT

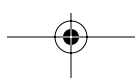
V

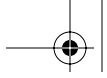
van Dam, Andries, xxi, 161, 165, 697
van Widenfelt, Rolf, 445
vendor-specific extensions, 587
versions, 585–587
 GLU, 586
vertex, 37
 also see vertex arrays
 evaluators, generating with, 518
 feedback mode, 575
 per-vertex operations pipeline stage, 12, 615
 specifying, 42
 tessellation, specifying for, 493, 501
 transformation pipeline, 100
vertex arrays, 66–82
 dereference a list of array elements, 74, 76, 77
 dereference a sequence of array elements, 78, 79
 dereference a single element, 73
 differences among releases, 29
 disabling, 69
 display list use, 283
 enabling, 68
 interleaved arrays, 80
 interleaved arrays, specifying, 81
 multitexturing texture coordinates, 436

 performance tips, 707
 querying, 621
 querying range values, 78
 reuse of vertices, 76
 sample program, 71, 77
 specifying data, 69
 steps to use, 67
 stride between data, 72, 81
video
 fake, 611
 flipping an image with `glPixelZoom()`, 329
 textured images, 376
viewing
 camera analogy, 100–101
viewing transformations, 104, 111, 120–125
 connection to modeling transformations, 105
 default position, 105
 different methods, 125
 pilot view, 126
 polar view, 127
 tripod analogy, 100
 up-vector, 105
viewing volume, 128
 clipping, 133, 144
 jittering, 475, 479
viewpoint
 lighting, for, 203
viewport transformations, 103, 108, 133–135
 photograph analogy, 100
 rendering pipeline stage, 12, 616
visual simulation
 fog, use of, 255
Voronoi polygons, 608

W

w coordinates, 38, 103, 108
 avoiding negative values, 705
 lighting, use with, 191
 perspective division, 136, 616
warping images, 606
Watt, Alan, 360





- web sites, xxi
 - errata list, xxiii
 - IBM OS/2 software and documentation, 674
 - Microsoft Developer Network, 677
 - WGL, 14, 677
 - wglCopyContext(), 678, 680
 - wglCreateContext(), 678, 680
 - wglCreateLayerContext(), 678, 680
 - wglDeleteContext(), 680
 - wglDescribeLayerPlane(), 678, 680
 - wglDestroyContext(), 678
 - wglGetCurrentContext(), 678, 681
 - wglGetCurrentDC(), 678, 681
 - wglGetLayerPaletteEntries(), 679, 681
 - wglGetProcAddress(), 681
 - wglMakeCurrent(), 678, 681
 - wglRealizeLayerPalette(), 679, 681
 - wglSetLayerPaletteEntries(), 681
 - wglShareLists(), 678, 680
 - wglSwapLayerBuffers(), 679, 681
 - wglUseFontBitmaps(), 680, 681
 - wglUseFontOutlines(), 680, 681
 - wglGetProcAddress(), 589
 - Williams, Lance, 390
 - Win32
 - ChoosePixelFormat(), 678, 680
 - CreateDIBitmap(), 679, 681
 - CreateDIBSection(), 679, 681
 - DeleteObject(), 679, 681
 - DescribePixelFormat(), 678, 680
 - GetVersion(), 678, 680
 - GetVersionEx(), 678, 680
 - SetPixelFormat(), 678, 680
 - SwapBuffers(), 679, 681
 - winding, 58
 - winding rules, 496–499
 - computational solid geometry, used for, 497
 - reversing winding direction, 500
 - window coordinates, 103, 133
 - feedback mode, 575
 - polygon offset, 268
 - raster position, 299
 - window management
 - glViewport() called, when window resized, 134
 - using GLUT, 17, 36
 - Windows, *See* Microsoft
 - working set of textures, 375, 403, 408
 - fragmentation of texture memory, 410
 - writemask, *See* masking (buffers)
 - writing pixel data, *See* pixel data (drawing)
 - www.opengl.org, xxi
- X**
- X Window System, 15, 662
 - client-server rendering, 5
 - minimum framebuffer configuration, 454
 - X Visual, 170, 661
- Z**
- z buffer, *See* depth buffer
 - z coordinates, *See* depth coordinates
 - zooming images, 328
 - filtered, 611

