

# Contents

<b>1</b>	<b>Getting Started</b>	<b>1</b>
1.1	Introduction	1
1.2	Access to Libraries	2
1.3	Data Types	2
1.4	Terminology	3
1.5	The OpenGL Utility Toolkit–GLUT	3
1.6	Our First Program	4
1.6.1	Drawing Primitives	7
1.6.2	Gaps, Lines and Blobs	10
1.6.3	Other Line Types	11
1.6.4	Surface Patterns and Polygon Stippling	12
1.6.5	Character Generation	15
1.7	Parametric Representation of Curves	17
1.7.1	The Growth or Logarithmic Spiral	19
1.8	Approximation and Processing Efficiency	21
1.8.1	Line Drawing	22
1.8.2	The Digital Difference Analyser	23
1.8.3	Bresenham’s Algorithm	25
1.8.4	Bresenham’s Algorithm for a Circle	27
1.9	Problem Solving Through Graphics	31
1.10	The Application Programming Interface (API)	34
1.10.1	The OpenGL Utility Toolkit (GLUT) Library	34
1.10.2	Mouse Interaction	35
1.10.3	Keyboard Interaction	36
1.11	Exercises	37
1.12	Conclusion	37
<b>2</b>	<b>Image File Formats</b>	<b>39</b>
2.1	Introduction	39
2.2	The Bitmap File	39
2.2.1	Bitmap-file Structure	40
2.2.2	A Typical File Layout	41

2.2.3	Reading and Drawing a Bitmap File .....	42
2.2.4	Creating a Display Window .....	45
2.2.5	Creating the Image for Display .....	46
2.2.6	Monochrome Bitmaps .....	46
2.2.7	A General Class for Input of Bit Map Files .....	47
2.2.8	Manipulating Bit Maps – Embossing .....	49
2.3	Audio/Video Interleaved (AVI) File Format .....	51
2.3.1	Chunk Hierarchy .....	53
2.3.2	An Example of Web Cam Display .....	55
2.3.3	Compression .....	58
2.4	Exercises .....	59
2.5	Conclusion .....	59
<b>3</b>	<b>Image Processing</b> .....	<b>61</b>
3.1	Introduction .....	61
3.2	Finding Objects .....	61
3.2.1	Edge Detection .....	62
3.2.2	Numerical Implementation .....	63
3.2.3	Sobel Operator .....	63
3.2.4	Convolution .....	64
3.2.5	Laplacian Operator .....	66
3.2.6	Smoothing and Edge Finding .....	69
3.3	Histogram Equalisation .....	71
3.4	Component (object) Labelling .....	73
3.4.1	Exploring Connectivity (1) .....	74
3.4.2	Exploring Connectivity (2) .....	76
3.4.3	Multiple Regions .....	77
3.5	Drawing on the Screen and Data Capture .....	77
3.5.1	Outlining of Areas of Interest on Images .....	80
3.5.2	An Approach to Automating Edge Detection .....	82
3.6	Exercises .....	85
3.7	Conclusion .....	85
<b>4</b>	<b>Transformations</b> .....	<b>87</b>
4.1	Introduction .....	87
4.2	Representing Points and Vectors .....	87
4.2.1	Vector Dot Products .....	89
4.2.2	The Vector Cross Product .....	89
4.2.3	Using the Cross Product to Provide Colour Variation to a Cylinder .....	91
4.3	Coordinate Systems .....	94
4.4	Transformations .....	95
4.4.1	Rotation .....	96
4.4.2	A Simple Example of Rotation .....	98

4.4.3	General Rotations About Any Point	99
4.5	The Model View Matrix	103
4.6	Adding Some Movement	103
4.6.1	Movement of Composite Objects	104
4.6.2	The Orrery	104
4.6.3	The Internal Combustion Engine	107
4.6.4	Arms, Robots, Actuators and Movement	113
4.6.5	Data Capture of Limb Movement	114
4.6.6	Animation – A Walking Robot	116
4.7	Collision Detection	120
4.8	Exercises	124
4.9	Conclusion	124
<b>5</b>	<b>Viewing and Projection</b>	<b>125</b>
5.1	Introduction	125
5.2	Vertex Pipeline	125
5.3	The Viewing Process	126
5.3.1	Engineering Drawings	128
5.4	Orthographic or Parallel Projection	132
5.4.1	From Real World to Screen Window	132
5.4.2	Inside the Matrix	134
5.4.3	Getting to Grips with Where Things Are	137
5.4.4	Perspective Projection	139
5.4.5	Understanding the View	143
5.5	Exercises	145
5.6	Conclusion	146
<b>6</b>	<b>Lighting and Colour</b>	<b>147</b>
6.1	Introduction	147
6.2	The Electromagnetic Spectrum	147
6.2.1	Colour	148
6.2.2	Mixing Colours	149
6.2.3	The Shading Model	150
6.3	Lighting and Illumination	151
6.3.1	Setting up the Lighting	152
6.3.2	Addition of Specular Light	157
6.4	A Synthetic Face	162
6.4.1	Getting the Vertex Coordinates	162
6.4.2	Moving the Eye and Light Source Together	163
6.5	Shadows – A Projection	164
6.5.1	Derivation of the Shadow Outline Using Equation of a Line	164
6.5.2	A Polygon Moving Shadow	166
6.6	Exercises	168
6.7	Conclusion	169

<b>7</b>	<b>Texture Mapping</b>	171
7.1	Introduction	171
7.2	The Mapping Process	171
7.3	Setting up Textures	172
7.3.1	Drawing the Texture	174
7.4	Mapping onto Curved Surfaces	176
7.4.1	Mercator Lines of Projection	176
7.4.2	Quadric Surfaces	177
7.4.3	Mapping onto a Spherical Surface	178
7.4.4	Mapping one Image to Sphere	182
7.4.5	Cylinder Texturing Mapping	184
7.4.6	Mapping to a Circle	185
7.4.7	Mapping to a Cone	186
7.4.8	Mapping onto a General Surface	189
7.4.9	Mapping onto a Contour Surface	191
7.5	Adding Some Movement – A Rotating Hemisphere	193
7.5.1	Movement via the Mouse	196
7.6	Using Bit Maps to Provide Context	200
7.6.1	Blending	202
7.6.2	Manipulating Pixels	206
7.7	Three-Dimensional Texturing	212
7.7.1	Simulation of a Spherical Object	213
7.7.2	Building a Solid Model of a Head	219
7.8	Exercises	226
7.9	Conclusion	226
<b>8</b>	<b>Objects to Artefacts</b>	227
8.1	Introduction	227
8.2	Artefact Hierarchy	227
8.2.1	Construction Double Sided Squares	228
8.2.2	A Box as a Basic Building Block	232
8.2.3	Building a Table	233
8.3	Integrating to a System	235
8.4	Changing the Furniture Style	243
8.5	Exercises	247
8.6	Conclusion	247
<b>9</b>	<b>Curves, Surfaces and Patterns</b>	249
9.1	Introduction	249
9.2	Fitting a Line or a Curve	250
9.2.1	Straight Line Fitting to Data Points	250
9.2.2	Lagrange Polynomials and Curve Fitting	251
9.2.3	A Practical Approach to Curve Fitting	251
9.2.4	Curve Fitting Using Cubic Splines	252
9.2.5	Surface Construction from Contour Maps	256

9.2.6	A Spline Surface Construction . . . . .	258
9.2.7	Bezier Curves . . . . .	260
9.2.8	General Curve Representation . . . . .	263
9.2.9	The Beginning's of a Bezier Surface . . . . .	267
9.2.10	Using Rotation to Sweep Out a Ring . . . . .	271
9.2.11	Patching a Surface . . . . .	271
9.3	Complex Shapes and Fractal Geometry . . . . .	274
9.3.1	Introductory Concepts . . . . .	275
9.3.2	Reduction Copying . . . . .	276
9.3.3	Implementation of a Sierpinski Gasket . . . . .	278
9.3.4	Pascal's Triangle . . . . .	280
9.3.5	The Sierpinski Gasket in 3 Dimensions . . . . .	280
9.3.6	Order Out of Chaos – A Non-deterministic Gasket? . . . . .	284
9.3.7	An Alternative Idea for a Tree or Plant . . . . .	290
9.4	Koch Curves . . . . .	292
9.4.1	Further Thoughts on the Koch Snowflake . . . . .	295
9.4.2	Self-Similarity . . . . .	296
9.4.3	Random Koch Fractals . . . . .	298
9.4.4	Back to Trees and Bushes . . . . .	300
9.5	Brownian Motion . . . . .	304
9.5.1	Fractalised Lines . . . . .	305
9.5.2	Fractal Surfaces . . . . .	307
9.6	Exercises . . . . .	310
9.7	Conclusion . . . . .	311
<b>Appendix A</b> . . . . .		313
A.1	Starting up VC++ 6.0 . . . . .	313
<b>Appendix B</b> . . . . .		317
B.1	Bresenham's Line Algorithm – General Conditions . . . . .	317
<b>Appendix C</b> . . . . .		319
C.1	Matrix and Vector Algebra . . . . .	319
C.2	Matrix Addition or Subtraction . . . . .	319
C.3	Matrix Multiplication . . . . .	319
C.4	Matrix Vector Multiplication . . . . .	320
C.5	Example of Matrix Multiplication . . . . .	321
C.6	Shearing Matrices . . . . .	321
<b>Appendix D</b> . . . . .		323
D.1	Equation of a Plane . . . . .	323
<b>References</b> . . . . .		325
<b>Index of Functions</b> . . . . .		327
<b>Index</b> . . . . .		329



<http://www.springer.com/978-1-84800-022-3>

OpenGL Graphics Through Applications

Whitrow, R.

2008, XV, 330 p., Softcover

ISBN: 978-1-84800-022-3