

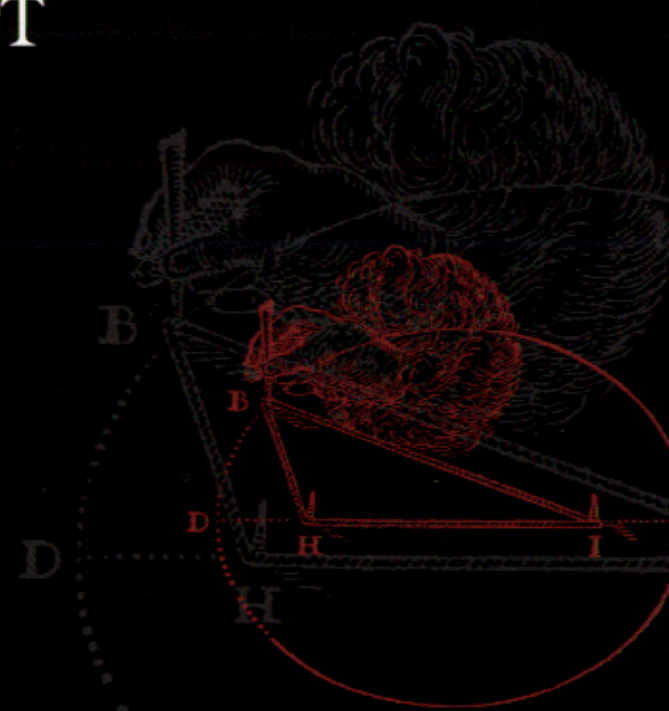


Bill Casselman

$$y = \int_{-\infty}^x e^{it^2} dt$$

MATHEMATICAL ILLUSTRATIONS

A MANUAL OF GEOMETRY
AND POSTSCRIPT



CAMBRIDGE

Contents

Preface	<i>page xi</i>
1 Getting started in PostScript	1
1.1. Simple drawing	1
1.2. Simple coordinate changes	7
1.3. Coordinate frames	9
1.4. Doing arithmetic in PostScript	11
1.5. Errors	14
1.6. Working with files and viewers GhostView or GSView	16
1.7. Some fine points	19
1.8. A trick for eliminating redundancy	22
1.9. Summary	23
1.10. Code	24
2 Elementary coordinate geometry	26
2.1. Points and vectors	26
2.2. Areas of parallelograms	27
2.3. Lengths	31
2.4. Vector projections	34
2.5. Rotations	37
2.6. The cosine rule	38
2.7. Dot products in higher dimensions	40
2.8. Lines	40
2.9. Code	43
3 Variables and procedures	44
3.1. Variables in PostScript	44
3.2. Procedures in PostScript	46
3.3. Keeping track of where you are	48
3.4. Passing arguments to procedures	50
3.5. Procedures as functions	52

3.6. Local variables	53
3.7. A final improvement	55
4 Coordinates and conditionals	57
4.1. Coordinates	57
4.2. How PostScript stores coordinate transformations	60
4.3. Picturing the coordinate system	63
4.4. Moving into three dimensions	65
4.5. How coordinate changes are made	69
4.6. Drawing infinite lines: Conditionals in PostScript	71
4.7. Another way to draw lines	76
4.8. Clipping	79
4.9. Order counts	79
4.10. Code	80
5 Drawing polygons: Loops and arrays	81
5.1. The repeat loop	81
5.2. The for loop	83
5.3. The loop loop	84
5.4. Graphing functions	84
5.5. General polygons	85
5.6. Clipping polygons	87
5.7. Code	91
6 Curves	92
6.1. Arcs	92
6.2. Fancier curves	93
6.3. Bézier curves	94
6.4. How to use Bézier curves	96
6.5. The mathematics of Bézier curves	103
6.6. Quadratic Bézier curves	105
6.7. Mathematical motivation	105
6.8. Weighted averages	108
6.9. How the computer draws Bézier curves	110
6.10. Bernstein polynomials	112
6.11. This section brings you the letter O	113
Interlude	117
7 Drawing curves automatically: Procedures as arguments	120
7.1. Drawing a hyperbola	120
7.2. Parametrized curves	125
7.3. Drawing graphs automatically	125
7.4. Drawing parametrized paths automatically	128

7.5. How to use it	130
7.6. How it works	132
7.7. Code	132
8 Nonlinear 2D transformations: Deconstructing paths	133
8.1. Two-dimensional transformations	133
8.2. Conformal transforms	137
8.3. Transforming paths	138
8.4. Maps	139
8.5. Stereographic projection	142
8.6. Fonts want to be free	144
8.7. Code	144
9 Recursion in PostScript	147
9.1. The perils of recursion	147
9.2. Sorting	149
9.3. Convex hulls	154
9.4. Code	159
10 Perspective and homogeneous coordinates	160
10.1. The projective plane	162
10.2. Boy's surface	164
10.3. Projective transformations	166
10.4. The fundamental theorem	167
10.5. Projective lines	169
10.6. A remark about solving linear systems	170
10.7. The GIMP perspective tool revisited	174
10.8. Projections in 2D	175
10.9. Perspective in 3D	175
11 Introduction to drawing in three dimensions	179
12 Transformations in 3D	181
12.1. Rigid transformations	181
12.2. Dot and cross products	183
12.3. Linear transformations and matrices	189
12.4. Changing coordinate systems	192
12.5. Rigid linear transformations	194
12.6. Orthogonal transformations in 2D	196
12.7. Orthogonal transformations in 3D	197
12.8. Calculating the effect of an axial rotation	200
12.9. Finding the axis and angle	201
12.10. Euler's theorem	202
12.11. More about projections	203

13 PostScript in 3D	205
13.1. A survey of the package	206
13.2. The 3D graphics environment	210
13.3. Coordinate transformations	212
13.4. Drawing	214
13.5. Surfaces	215
13.6. Code	216
14 Drawing surfaces in 3D	217
14.1. Faces	217
14.2. Polyhedra	219
14.3. Visibility for convex polyhedra	221
14.4. Shading	223
14.5. Smooth surfaces	227
14.6. Smoother surfaces	231
14.7. Abandoning convexity	236
14.8. Summary	240
14.9. Code	241
15 Triangulation: Basic graphics algorithms	243
15.1. The monotone decomposition	243
15.2. The algorithm	248
15.3. The intersection list	250
15.4. Triangulation	252
15.5. Small triangles	254
15.6. Code	256
Appendix 1. Summary of PostScript commands	259
A1.1. Mathematical functions	259
A1.2. Stack operations	260
A1.3. Arrays	260
A1.4. Dictionaries	261
A1.5. Conditionals	262
A1.6. Loops	262
A1.7. Conversions	263
A1.8. File handling and miscellaneous	263
A1.9. Display	264
A1.10. Graphics state	265
A1.11. Coordinates	266
A1.12. Drawing	266
A1.13. Displaying text	267
A1.14. Errors	268
A1.15. Alphabetical list	269

Appendix 2. Setting up your PostScript environment	271
A2.1. Editing PostScript files	271
A2.2. Running external files	272
A2.3. Making images	272
A2.4. Printing files	273
Appendix 3. Structured PostScript documents	276
Appendix 4. Simple text display	279
A4.1. <i>Simple PostScript text</i>	279
A4.2. Outline fonts	281
Appendix 5. Zooming	283
A5.1. Zooming	283
A5.2. An explicit procedure	285
A5.3. Playing around	285
A5.4. Code	286
Appendix 6. Evaluating polynomials: Getting along without variables	287
A6.1. The most straightforward way to do it	287
A6.2. Horner's method	288
A6.3. Evaluating the derivatives efficiently	290
A6.4. Evaluating Bernstein polynomials	291
A6.5. Code	293
Appendix 7. Importing PostScript files	294
A7.1. <i>Labeling a graph</i>	294
A7.2. Importing $\TeX$ text	298
A7.3. Fancy work	301
Epilogue	305
Index	313