

markiii

Claude Heiland-Allen

2010–2019

Contents

1	borgan~.pd	2
2	cm-ead~.pd	5
3	compress~.pd	6
4	geopipe-help.pd	7
5	geo-pipe.pd	8
6	geo-pipes2.pd	10
7	geo-pipes.pd	11
8	hihat~.pd	11
9	hverb-calc.pd_lua	12
10	hverb~-help.pd	13
11	hverb~.pd	15
12	keyboard-input.pd	20
13	keyboard-mapping.pd_lua	20
14	markiiiflex2-jack.sh	22
15	markiiiflex2-oss.sh	22
16	markiiiflex2.pd	22
17	markiiiflex2.sh	32
18	markiiiflex-jack.sh	32
19	markiiiflex-oss.sh	32
20	markiiiflex.pd	32
21	markiiiflex.sh	36
22	markiii-help-old.pd	36
23	markiii.pd_lua	38
24	markiii-sonify.pd	40
25	markiii-visualize.pd	41
26	recorder~.pd	44
27	rhythmiii.pd	45
28	snare~.pd	52
29	sphere-segments2.pd_lua	53
30	sphere-segments.pd_lua	62

1 borgan~.pd

```

#N canvas 300 86 722 583 10;
#X obj 38 30 r pd;
#X obj 38 52 route dsp;
#X obj 38 73 select 1;
5 #X obj 132 129 samplerate~;
#X obj 132 177 t f f;
#X obj 11 10 loadbang;
#X obj 162 323 noise~;

```

```

#X obj 133 390 hip~ 10;
10 #X obj 131 433 lop~ 1000;
#X obj 131 454 lop~ 1000;
#X obj 11 194 metro;
#X obj 132 411 hip~ 10;
#X obj 41 150 /;
15 #X obj 11 96 t b b b;
#X text 24 567 borgan~ (copyleft) claude@mathr.co.uk;
#X obj 44 452 osc~;
#X obj 43 475 *~;
#X obj 26 426 * 0.5;
20 #X obj 302 197 0;
#X obj 42 129 1.024e+06;
#X obj 67 427 -0.25;
#X obj 132 265 -~;
#X obj 132 367 expr~ tanh($v1*1.5);
25 #X obj 132 153 / 1024;
#X obj 162 345 *~ 0.001;
#X obj 131 541 table \ $0-osc-1 1027;
#X obj 132 220 tabosc4~ \ $0-osc-1;
#X obj 162 241 tabosc4~ \ $0-osc-1;
30 #X obj 132 517 tabwrite~ \ $0-osc-1;
#X obj 214 50 tabosc4~ \ $0-osc-1;
#X obj 282 541 table \ $0-osc-2 1027;
#X obj 344 50 tabosc4~ \ $0-osc-2;
#X obj 213 117 *~;
35 #X obj 343 117 *~;
#X obj 282 517 tabwrite~ \ $0-osc-2;
#X obj 378 174 f 0;
#X obj 413 173 + 1;
#X obj 414 195 mod 2;
40 #X obj 374 226 sel 0 1;
#X obj 273 99 vline~;
#X obj 359 97 -~;
#X obj 359 76 sig~ 1;
#X obj 282 472 delay 5;
45 #X obj 131 475 delay 5;
#X msg 273 77 \ $1 5;
#X obj 162 300 -~;
#X obj 132 323 *~;
#X obj 142 343 *~;
50 #X obj 192 281 tabosc4~ \ $0-osc-2;
#X obj 162 260 tabosc4~ \ $0-osc-2;
#X obj 162 199 * -2;
#X obj 480 164 *~;
#X obj 493 51 sel 0;
55 #X msg 494 96 0 10;
#X obj 495 123 vline~;
#X obj 289 18 vline~;
#X obj 480 203 outlet~;
#X msg 507 75 1 10;
60 #X obj 492 28 inlet;
#X connect 0 0 1 0;
#X connect 1 0 2 0;
#X connect 2 0 13 0;
#X connect 3 0 12 1;
65 #X connect 3 0 23 0;

```

```
#X connect 4 0 26 0;
#X connect 4 0 49 0;
#X connect 4 1 50 0;
#X connect 5 0 13 0;
70 #X connect 6 0 24 0;
#X connect 7 0 11 0;
#X connect 8 0 9 0;
#X connect 9 0 16 1;
#X connect 10 0 35 0;
75 #X connect 11 0 8 0;
#X connect 12 0 10 1;
#X connect 13 0 10 0;
#X connect 13 1 19 0;
#X connect 13 2 3 0;
80 #X connect 15 0 16 0;
#X connect 16 0 28 0;
#X connect 16 0 34 0;
#X connect 17 0 15 0;
#X connect 18 0 26 1;
85 #X connect 18 0 27 1;
#X connect 18 0 48 1;
#X connect 18 0 49 1;
#X connect 19 0 12 0;
#X connect 20 0 15 1;
90 #X connect 21 0 46 0;
#X connect 22 0 7 0;
#X connect 23 0 4 0;
#X connect 23 0 17 0;
#X connect 24 0 22 0;
95 #X connect 26 0 21 0;
#X connect 27 0 21 1;
#X connect 29 0 32 0;
#X connect 31 0 33 0;
#X connect 32 0 51 0;
100 #X connect 33 0 51 0;
#X connect 35 0 36 0;
#X connect 35 0 38 0;
#X connect 35 0 44 0;
#X connect 36 0 37 0;
105 #X connect 37 0 35 1;
#X connect 38 0 43 0;
#X connect 38 1 42 0;
#X connect 39 0 32 1;
#X connect 39 0 40 1;
110 #X connect 39 0 46 1;
#X connect 40 0 33 1;
#X connect 40 0 47 1;
#X connect 41 0 40 0;
#X connect 42 0 34 0;
115 #X connect 42 0 20 0;
#X connect 42 0 18 0;
#X connect 43 0 28 0;
#X connect 43 0 20 0;
#X connect 43 0 18 0;
120 #X connect 44 0 39 0;
#X connect 45 0 47 0;
#X connect 46 0 22 0;
```

```

#X connect 47 0 22 0;
#X connect 48 0 45 1;
125 #X connect 49 0 45 0;
#X connect 50 0 27 0;
#X connect 50 0 48 0;
#X connect 51 0 56 0;
#X connect 52 0 53 0;
130 #X connect 52 1 57 0;
#X connect 52 1 55 0;
#X connect 53 0 54 0;
#X connect 54 0 51 1;
#X connect 55 0 29 0;
135 #X connect 55 0 31 0;
#X connect 57 0 54 0;
#X connect 58 0 52 0;

```

2 cm-ead~.pd

```

#N canvas 0 0 450 300 10;
#X obj 21 17 inlet;
#X obj 81 17 inlet;
#X obj 141 17 inlet;
5 #X obj 313 56 samplerate~;
#X obj 182 5 loadbang;
#X obj 182 32 t b b;
#X obj 141 68 f \$2;
#X obj 81 84 f \$1;
10 #X obj 97 201 vline~;
#X obj 66 157 f;
#X obj 126 159 f;
#X obj 81 137 clip 0 1;
#X obj 141 138 clip 0 1;
15 #X obj 44 237 rpole~;
#X obj 21 107 delay;
#X obj 21 202 vline~;
#X obj 21 155 f 0;
#X obj 44 262 outlet~;
20 #X obj 42 157 f 1;
#X obj 141 88 expr exp((1000*log(0.001))/($f1*$f2));
#X obj 81 118 expr exp((1000*log(0.001))/($f1*$f2));
#X obj 47 180 -;
#X obj 21 45 t b b b b b b;
25 #X connect 0 0 22 0;
#X connect 1 0 14 1;
#X connect 1 0 7 0;
#X connect 2 0 6 0;
#X connect 3 0 19 1;
30 #X connect 3 0 20 1;
#X connect 4 0 5 0;
#X connect 5 0 6 0;
#X connect 5 0 7 0;
#X connect 5 1 3 0;
35 #X connect 6 0 19 0;
#X connect 7 0 14 1;
#X connect 7 0 20 0;
#X connect 8 0 13 1;
#X connect 9 0 8 0;

```

```

40  #X connect 9 0 21 1;
    #X connect 10 0 8 0;
    #X connect 11 0 9 1;
    #X connect 12 0 10 1;
    #X connect 13 0 17 0;
45  #X connect 14 0 10 0;
    #X connect 14 0 16 0;
    #X connect 15 0 13 0;
    #X connect 16 0 15 0;
    #X connect 18 0 21 0;
50  #X connect 19 0 12 0;
    #X connect 20 0 11 0;
    #X connect 21 0 15 0;
    #X connect 22 0 14 0;
    #X connect 22 1 18 0;
55  #X connect 22 2 9 0;
    #X connect 22 3 7 0;
    #X connect 22 4 6 0;
    #X connect 22 5 3 0;

```

3 compress~.pd

```

#N canvas 275 79 452 587 10;
#X obj 28 28 inlet ~;
#X obj 176 20 inlet ~;
#X obj 27 543 outlet ~;
5  #X obj 177 544 outlet ~;
    #X obj 58 71 hip~ 5;
    #X obj 121 71 hip~ 5;
    #X obj 58 97 *~;
    #X obj 120 98 *~;
10  #X obj 87 158 +~;
    #X obj 281 25 inlet;
    #X obj 87 281 rmstodb~;
    #X obj 86 353 dbtorms~;
    #X obj 27 474 *~;
15  #X obj 177 474 *~;
    #X obj 120 123 lop~ 10;
    #X obj 57 124 lop~ 10;
    #X obj 87 221 lop~ 25;
    #X obj 25 272 /~;
20  #X obj 178 270 /~;
    #X obj 201 390 dbtorms;
    #X obj 86 417 /~ 0;
    #X obj 87 179 sqrt~;
    #X obj 86 442 *~ 0.25;
25  #X obj 310 55 loadbang;
    #X obj 310 79 f \$1;
    #X obj 27 514 expr~ tanh($v1);
    #X obj 177 514 expr~ tanh($v1);
    #X obj 86 314 expr~ if($v1>$f2 \, $f2+($v1-$f2)/8 \, $f2);
30  #X obj 201 366 expr (100-$f1)/8+$f1;
    #X obj 88 202 +~ 0.01;
    #X connect 0 0 4 0;
    #X connect 0 0 17 0;
    #X connect 1 0 5 0;
35  #X connect 1 0 18 0;

```

```

#X connect 4 0 6 0;
#X connect 4 0 6 1;
#X connect 5 0 7 0;
#X connect 5 0 7 1;
40 #X connect 6 0 15 0;
#X connect 7 0 14 0;
#X connect 8 0 21 0;
#X connect 9 0 27 1;
#X connect 9 0 28 0;
45 #X connect 10 0 27 0;
#X connect 11 0 20 0;
#X connect 12 0 25 0;
#X connect 13 0 26 0;
#X connect 14 0 8 1;
50 #X connect 15 0 8 0;
#X connect 16 0 17 1;
#X connect 16 0 18 1;
#X connect 16 0 10 0;
#X connect 17 0 12 0;
55 #X connect 18 0 13 0;
#X connect 19 0 20 1;
#X connect 20 0 22 0;
#X connect 21 0 29 0;
#X connect 22 0 12 1;
60 #X connect 22 0 13 1;
#X connect 23 0 24 0;
#X connect 24 0 27 1;
#X connect 24 0 28 0;
#X connect 25 0 2 0;
65 #X connect 26 0 3 0;
#X connect 27 0 11 0;
#X connect 28 0 19 0;
#X connect 29 0 16 0;

```

4 geopipe-help.pd

```

#N canvas 11 273 450 422 10;
#X obj 134 147 gemwin 25;
#X obj 116 173 gemhead;
#X msg 198 300 0.1;
5 #X obj 117 194 t a a;
#X obj 168 222 world_light;
#X msg 228 198 1 0 1;
#X msg 136 87 create \, 1 \, lighting 0;
#X msg 189 267 1 1 0;
10 #X obj 145 365 geo-pipe;
#X floatatom 266 231 5 0 0 0 - - -;
#X obj 296 284 pack f f f;
#X obj 306 259 t b a;
#X floatatom 312 230 5 0 0 0 - - -;
15 #X obj 346 259 t b a;
#X floatatom 352 230 5 0 0 0 - - -;
#X obj 283 312 t a b;
#X msg 296 334 -2 -2 0;
#X connect 1 0 3 0;
20 #X connect 2 0 8 3;
#X connect 3 0 8 0;

```

```

#X connect 3 1 4 0;
#X connect 5 0 4 1;
#X connect 6 0 0 0;
25 #X connect 7 0 8 2;
#X connect 9 0 10 0;
#X connect 10 0 15 0;
#X connect 11 0 10 0;
#X connect 11 1 10 1;
30 #X connect 12 0 11 0;
#X connect 13 0 10 0;
#X connect 13 1 10 2;
#X connect 14 0 13 0;
#X connect 15 0 8 1;
35 #X connect 15 1 16 0;
#X connect 16 0 8 2;

```

5 geo-pipe.pd

```

#N canvas 278 308 450 387 10;
#X obj 15 19 inlet;
#X obj 113 19 inlet;
#X obj 211 19 inlet;
5 #X obj 323 50 inlet;
#X obj 16 40 t a a;
#X obj 16 361 outlet;
#X obj 113 44 unpack f f f;
#X obj 211 44 unpack f f f;
10 #X obj 173 102 -;
#X obj 203 102 -;
#X obj 43 62 separator;
#X obj 225 262 loadbang;
#X obj 30 318 cylinder;
15 #X obj 172 150 t f f;
#X obj 173 174 *;
#X obj 44 98 translate 1 0 0 0;
#X obj 46 223 rotate;
#X obj 212 150 t f f;
20 #X obj 213 174 *;
#X obj 262 150 t f f;
#X obj 263 174 *;
#X obj 218 199 +;
#X obj 174 202 +;
25 #X obj 174 225 sqrt;
#X obj 174 248 swap;
#X obj 175 272 /;
#X obj 174 299 expr 180.0*acos($f1)/3.1415926;
#X obj 174 329 *;
30 #X obj 211 317 expr if($f1<0 \, -1 \, 1);
#X obj 93 196 pack f f;
#X msg 93 221 \"$2 \"$1 0;
#X obj 138 174 * -1;
#X obj 139 154 *;
35 #X obj 93 156 *;
#X obj 32 293 translateXYZ 0 0 0;
#X obj 235 104 +;
#X obj 248 66 -;
#X obj 311 101 == 0;

```



```
40  #X obj 313 121 * 1e-05;
    #X obj 250 87 t f f;
    #X obj 149 228 /;
    #X obj 142 252 * 0.5;
    #X obj 31 271 scaleXYZ -1 1 1;
45  #X obj 67 360 sphere;
    #X obj 65 341 separator;
    #X obj 35 249 separator;
    #X msg 224 280 7;
    #X connect 0 0 4 0;
50  #X connect 1 0 6 0;
    #X connect 2 0 7 0;
    #X connect 2 0 15 2;
    #X connect 3 0 12 1;
    #X connect 3 0 34 3;
55  #X connect 3 0 40 1;
    #X connect 3 0 43 1;
    #X connect 4 0 5 0;
    #X connect 4 1 10 0;
    #X connect 6 0 8 0;
60  #X connect 6 1 9 0;
    #X connect 6 2 36 0;
    #X connect 7 0 8 1;
    #X connect 7 1 9 1;
    #X connect 7 2 36 1;
65  #X connect 8 0 13 0;
    #X connect 8 0 33 0;
    #X connect 9 0 17 0;
    #X connect 9 0 32 0;
    #X connect 10 0 15 0;
70  #X connect 11 0 46 0;
    #X connect 13 0 14 0;
    #X connect 13 1 14 1;
    #X connect 14 0 22 0;
    #X connect 15 0 16 0;
75  #X connect 16 0 45 0;
    #X connect 16 0 44 0;
    #X connect 17 0 18 0;
    #X connect 17 1 18 1;
    #X connect 18 0 21 0;
80  #X connect 19 0 20 0;
    #X connect 19 1 20 1;
    #X connect 20 0 21 1;
    #X connect 21 0 22 1;
    #X connect 22 0 23 0;
85  #X connect 23 0 24 0;
    #X connect 23 0 40 0;
    #X connect 24 0 25 0;
    #X connect 24 1 25 1;
    #X connect 25 0 26 0;
90  #X connect 26 0 27 0;
    #X connect 27 0 16 1;
    #X connect 28 0 27 1;
    #X connect 29 0 30 0;
    #X connect 30 0 16 2;
95  #X connect 31 0 29 1;
    #X connect 32 0 31 0;
```

```

#X connect 33 0 29 0;
#X connect 34 0 12 0;
#X connect 35 0 19 0;
100 #X connect 35 0 24 1;
#X connect 35 0 28 0;
#X connect 35 0 32 1;
#X connect 35 0 33 1;
#X connect 36 0 39 0;
105 #X connect 37 0 38 0;
#X connect 38 0 35 1;
#X connect 39 0 35 0;
#X connect 39 1 37 0;
#X connect 40 0 41 0;
110 #X connect 41 0 42 3;
#X connect 42 0 34 0;
#X connect 44 0 43 0;
#X connect 45 0 42 0;
#X connect 46 0 12 2;
115 #X connect 46 0 43 2;

```

6 geo-pipes2.pd

```

#N canvas 0 0 608 346 10;
#X obj 88 97 t b a;
#X obj 34 118 gemlist;
#X obj 50 63 inlet;
5 #X obj 88 66 inlet;
#X obj 156 65 inlet;
#X obj 206 65 inlet;
#X obj 259 66 inlet;
#X obj 330 65 inlet;
10 #X obj 8 63 inlet;
#X obj 330 136 GEMglBegin;
#X obj 389 91 GLdefine GL_LINES;
#X obj 390 66 loadbang;
#X obj 6 94 GEMglEnd;
15 #X obj 34 204 GEMglVertex3fv;
#X obj 34 279 GEMglVertex3fv;
#X obj 34 182 GEMglNormal3fv;
#X msg 420 181 0 0 1;
#X obj 34 257 GEMglNormal3fv;
20 #X obj 34 231 GEMglColor3fv;
#X obj 34 158 GEMglColor3fv;
#X connect 0 0 1 0;
#X connect 0 1 13 1;
#X connect 1 0 19 0;
25 #X connect 2 0 1 1;
#X connect 3 0 0 0;
#X connect 4 0 14 1;
#X connect 6 0 18 1;
#X connect 7 0 9 0;
30 #X connect 8 0 12 0;
#X connect 10 0 9 1;
#X connect 11 0 10 0;
#X connect 11 0 16 0;
#X connect 13 0 18 0;
35 #X connect 15 0 13 0;

```

```

#X connect 16 0 15 1;
#X connect 16 0 17 1;
#X connect 17 0 14 0;
#X connect 18 0 17 0;
40 #X connect 19 0 15 0;

```

7 geo-pipes.pd

```

#N canvas 0 0 450 300 10;
#X obj 88 97 t b a;
#X obj 34 118 gemlist;
#X obj 34 190 geo-pipe;
5 #X obj 34 158 color;
#X obj 50 63 inlet;
#X obj 88 66 inlet;
#X obj 156 65 inlet;
#X obj 206 65 inlet;
10 #X obj 259 66 inlet;
#X obj 330 65 inlet;
#X obj 8 63 inlet;
#X connect 0 0 1 0;
#X connect 0 1 2 1;
15 #X connect 1 0 3 0;
#X connect 3 0 2 0;
#X connect 4 0 1 1;
#X connect 5 0 0 0;
#X connect 6 0 2 2;
20 #X connect 7 0 2 3;
#X connect 8 0 3 1;

```

8 hihat~.pd

```

#N canvas 0 0 459 480 10;
#X obj 140 7 inlet;
#X obj 202 400 outlet~;
#X obj 83 181 *~;
5 #X obj 230 183 *~;
#X obj 150 213 *~;
#X obj 139 265 +~;
#X obj 150 300 osc~ 5435;
#X obj 150 236 *~ 12000;
10 #X obj 23 144 osc~ 6221;
#X obj 248 142 osc~ 2441;
#X obj 228 235 noise~;
#X obj 180 328 +~;
#X obj 230 299 clip~ -0.7 0.7;
15 #X obj 180 354 *~;
#X obj 235 352 *~;
#X obj 245 373 *~ 0.06;
#X obj 177 374 *~ 0.06;
#X obj 78 238 sig~ 7435;
20 #X obj 228 256 bp~ 12000 12;
#X obj 172 143 osc~ 3283;
#X obj 98 144 osc~ 4109;
#X obj 245 111 sqrt~;
#X obj 246 87 cm-ead~ 2 800;
25 #X obj 139 89 cm-ead~ 2 200;

```

```

#X obj 157 117 cm-ead~ 2 100;
#X obj 139 65 t b f;
#X obj 139 45 moses 48;
#X obj 199 62 * 4;
30 #X obj 239 55 t b f;
#X obj 321 64 * 8;
#X obj 140 26 sel 0;
#X connect 0 0 30 0;
#X connect 2 0 4 0;
35 #X connect 3 0 4 1;
#X connect 4 0 7 0;
#X connect 5 0 6 0;
#X connect 6 0 11 0;
#X connect 7 0 5 1;
40 #X connect 8 0 2 0;
#X connect 9 0 3 1;
#X connect 10 0 18 0;
#X connect 11 0 13 0;
#X connect 11 0 14 0;
45 #X connect 12 0 11 1;
#X connect 13 0 16 0;
#X connect 14 0 15 0;
#X connect 15 0 1 0;
#X connect 16 0 1 0;
50 #X connect 17 0 5 0;
#X connect 18 0 12 0;
#X connect 19 0 3 0;
#X connect 20 0 2 1;
#X connect 21 0 13 1;
55 #X connect 22 0 21 0;
#X connect 23 0 13 1;
#X connect 24 0 14 1;
#X connect 25 0 23 0;
#X connect 25 0 24 0;
60 #X connect 25 1 27 0;
#X connect 26 0 25 0;
#X connect 26 1 28 0;
#X connect 27 0 23 2;
#X connect 28 0 22 0;
65 #X connect 28 0 24 0;
#X connect 28 1 29 0;
#X connect 29 0 22 2;
#X connect 30 1 26 0;

```

9 hverb-calc.pd_lua

```

local R = pd.Class:new():register("hverb-calc")

function R:initialize(sel, atoms)
  if type(atoms[1]) ~= "number" then
5     return false
  end
  self.count = math.max(atoms[1], 1)
  self.inlets = 1
  self.outlets = 1
10   return true
end

```

```

-- note: gcd(0,0) == 0, which is what we want, if not math-correct
local gcd = function(m, n)
15   while m ~= 0 do m, n = math.mod(n, m), m end
   return n
end

-- precalculate fundamental modes
20 local fundamentals = { }
   for nx = 0, 16 do
     for ny = 0, 16 do
       for nz = 0, 16 do
         for nw = 0, 16 do
25           if gcd(gcd(gcd(nx, ny), nz), nw) == 1 then
               table.insert(fundamentals, { nx, ny, nz, nw })
             end
           end end end end

30 function R:delay(lx, ly, lz, lw, nx, ny, nz, nw)
   return 1000 / ((340/2) * math.sqrt((nx*nx)/(lx*lx)+(ny*ny)/(ly*ly)+(nz*nz)/(lz*
     ↵ *lz)+(nw*nw)/(lw*lw)))
end

function R:in_l_list(atoms)
35   local lx = atoms[1]
   local ly = atoms[2]
   local lz = atoms[3]
   local lw = atoms[4]
   local delays = { }
40   for i,f in ipairs(fundamentals) do
       table.insert(delays, self:delay(lx, ly, lz, lw, unpack(f)))
     end
   table.sort(delays, function(a, b) return a > b end)
   local out = { }
45   local j = 1
   for i = 1, self.count do
       out[i] = delays[j]
       repeat
         j = j + 1
50       until delays[j] ~= delays[j-1]
     end
   self:outlet(1, "list", out)
end

```

10 hverb~-help.pd

```

#N canvas 0 0 585 522 10;
#X obj 62 354 hverb~;
#X obj 11 103 noise~;
#X obj 32 210 *~;
5 #X obj 62 134 vline~;
#X obj 90 253 hverb-calc 16;
#X floatatom 134 134 5 0 0 0 - - -;
#X obj 282 283 heref-calc;
#X obj 135 321 heref~;
10 #X obj 202 283 heref-calc;
#X obj 302 259 t a a;

```

```

#X msg 306 236 0.5 0.5 0.5 0.5;
#X msg 285 213 0.25 0.25 0.25 0.25;
#X msg 207 193 0.26 0.24 0.26 0.24;
15 #X obj 233 358 print heref~L;
#X obj 305 335 print heref~R;
#X obj 24 313 heref~;
#X obj 157 201 / 100;
#X floatatom 158 172 5 0 0 0 - - -;
20 #X msg 61 108 0.5 1 \, 0 1 1;
#X obj 62 168 *~;
#X obj 209 107 + 1;
#X obj 209 83 random 16;
#X obj 279 107 + 1;
25 #X obj 279 83 random 16;
#X obj 349 107 + 1;
#X obj 349 83 random 16;
#X obj 419 107 + 1;
#X obj 419 83 random 16;
30 #X obj 269 144 pack f f f f;
#X obj 273 51 t b b b b;
#X obj 112 42 t b b b;
#X obj 113 17 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144 -1
-1;
35 #X obj 391 167 print room;
#X obj 269 169 t b b b a a a a;
#X obj 152 381 print hverb~;
#X obj 112 -5 spigot;
#X obj 132 18 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1
40 1;
#X obj 37 27 delay;
#X obj 62 417 dac~;
#X obj 33 385 *~ 1;
#X obj 101 384 *~ 1;
45 #X obj 57 461 writesf~ 2;
#X msg 157 417 open -bytes 4 hverb.wav \, start;
#X msg 155 442 stop;
#X obj 135 64 random 3000;
#X obj 134 84 + 30;
50 #X connect 0 0 39 0;
#X connect 0 1 40 0;
#X connect 1 0 2 0;
#X connect 2 0 7 0;
#X connect 2 0 15 0;
55 #X connect 3 0 19 0;
#X connect 3 0 19 1;
#X connect 4 0 0 2;
#X connect 4 0 34 0;
#X connect 5 0 0 3;
60 #X connect 6 0 7 1;
#X connect 6 0 14 0;
#X connect 7 0 40 0;
#X connect 7 1 0 1;
#X connect 8 0 13 0;
65 #X connect 8 0 15 1;
#X connect 9 0 8 1;
#X connect 9 1 6 1;
#X connect 10 0 9 0;

```

```

#X connect 11 0 6 0;
70 #X connect 12 0 8 0;
#X connect 15 0 39 0;
#X connect 15 1 0 0;
#X connect 16 0 7 2;
#X connect 16 0 15 2;
75 #X connect 17 0 16 0;
#X connect 18 0 3 0;
#X connect 19 0 2 1;
#X connect 20 0 28 0;
#X connect 21 0 20 0;
80 #X connect 22 0 28 1;
#X connect 23 0 22 0;
#X connect 24 0 28 2;
#X connect 25 0 24 0;
#X connect 26 0 28 3;
85 #X connect 27 0 26 0;
#X connect 28 0 33 0;
#X connect 29 0 21 0;
#X connect 29 1 23 0;
#X connect 29 2 25 0;
90 #X connect 29 3 27 0;
#X connect 30 0 18 0;
#X connect 30 1 44 0;
#X connect 30 2 29 0;
#X connect 31 0 30 0;
95 #X connect 33 0 12 0;
#X connect 33 1 11 0;
#X connect 33 2 10 0;
#X connect 33 3 4 0;
#X connect 33 4 8 2;
100 #X connect 33 5 6 2;
#X connect 33 6 32 0;
#X connect 35 0 31 0;
#X connect 36 0 35 1;
#X connect 37 0 35 0;
105 #X connect 39 0 38 0;
#X connect 39 0 41 0;
#X connect 40 0 38 1;
#X connect 40 0 41 1;
#X connect 42 0 41 0;
110 #X connect 43 0 41 0;
#X connect 44 0 45 0;
#X connect 45 0 5 0;
#X connect 45 0 37 0;

```

11 hverb~.pd

```

#N canvas 0 0 967 676 10;
#X obj 152 273 mtx_*~ 16 16 0 .....;
#X obj 198 80 delread~ \$0-0;
#X obj 208 100 delread~ \$0-1;
5 #X obj 218 120 delread~ \$0-2;
#X obj 228 140 delread~ \$0-3;
#X obj 238 160 delread~ \$0-4;
#X obj 248 180 delread~ \$0-5;
#X obj 258 200 delread~ \$0-6;

```

```

10  #X obj 268 220 delread~ \$0-7;
    #X obj 298 80 delread~ \$0-8;
    #X obj 308 100 delread~ \$0-9;
    #X obj 318 120 delread~ \$0-10;
    #X obj 328 140 delread~ \$0-11;
15  #X obj 338 160 delread~ \$0-12;
    #X obj 348 180 delread~ \$0-13;
    #X obj 358 200 delread~ \$0-14;
    #X obj 368 220 delread~ \$0-15;
    #X obj 222 55 unpack f f f f f f f f f f f f f f f f;
20  #X obj 189 10 inlet;
    #X obj 166 243 mtx.*~ 16 2 0 .....;
    #X obj 78 11 inlet~;
    #X obj 128 11 inlet~;
    #X obj 266 6 inlet;
25  #X msg 513 46 matrix 16 2 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0
    1 0 1 0 1 0 1 0 1 0 1;
    #X obj 551 9 loadbang;
    #X obj 513 96 mtx../ 8;
    #X obj 141 322 outlet~;
30  #X obj 375 321 outlet~;
    #X obj 506 205 mtx../ 8;
    #X msg 572 242 matrix 2 16 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 1 1 1 1 1 1 1 1;
    #X msg 581 84 matrix 16 16 1 -1 -1 -1 -1 1 1 1 -1 1 1 1 -1 1 1 1 -1
35  1 -1 -1 1 -1 1 1 1 -1 1 1 1 -1 1 1 -1 1 1 -1 1 1 1 -1 1 1 1
    -1 1 -1 -1 -1 1 1 1 1 -1 1 1 1 -1 1 1 1 1 -1 -1 -1 -1 1
    1 1 -1 1 1 1 1 -1 1 1 -1 1 -1 -1 1 1 -1 1 1 1 1 -1 1 -1 -1 1
    -1 1 1 -1 1 1 1 -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 1 1 1 1
    -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 1 1 1 -1 1 -1 -1 1 1
40  1 1 -1 1 1 1 -1 1 -1 -1 1 -1 1 1 1 -1 1 1 1 -1 -1 -1 -1 1
    1 1 1 -1 -1 1 1 1 -1 1 1 1 -1 1 1 1 1 -1 -1 -1 1 -1 1 1 1
    -1 1 1 -1 1 -1 -1 1 1 -1 1 1 1 -1 1 1 1 -1 1 -1 -1 1 1 1 -1 1
    1 1 -1 1 1 1 -1 -1 -1 1;
    #X obj 508 179 mtx../ 4;
45  #X obj 114 565 delwrite~ \$0-0 1000;
    #X obj 124 585 delwrite~ \$0-1 1000;
    #X obj 134 605 delwrite~ \$0-2 1000;
    #X obj 144 625 delwrite~ \$0-3 1000;
    #X obj 154 645 delwrite~ \$0-4 1000;
50  #X obj 164 665 delwrite~ \$0-5 1000;
    #X obj 174 685 delwrite~ \$0-6 1000;
    #X obj 184 705 delwrite~ \$0-7 1000;
    #X obj 254 565 delwrite~ \$0-8 1000;
    #X obj 264 585 delwrite~ \$0-9 1000;
55  #X obj 274 605 delwrite~ \$0-10 1000;
    #X obj 284 625 delwrite~ \$0-11 1000;
    #X obj 294 645 delwrite~ \$0-12 1000;
    #X obj 304 665 delwrite~ \$0-13 1000;
    #X obj 314 685 delwrite~ \$0-14 1000;
60  #X obj 324 705 delwrite~ \$0-15 1000;
    #X obj 139 301 mtx.*~ 2 16 0 .....;
    #X obj 112 521 *~ 0;
    #X obj 152 521 *~ 0;
    #X obj 192 521 *~ 0;
65  #X obj 232 521 *~ 0;
    #X obj 272 521 *~ 0;

```



```

#X obj 312 521 *~ 0;
#X obj 352 521 *~ 0;
#X obj 392 521 *~ 0;
70 #X obj 132 541 *~ 0;
#X obj 172 541 *~ 0;
#X obj 212 541 *~ 0;
#X obj 252 541 *~ 0;
#X obj 292 541 *~ 0;
75 #X obj 332 541 *~ 0;
#X obj 372 541 *~ 0;
#X obj 412 541 *~ 0;
#X obj 213 356 expr pow(10 \, -3*$f2/$f1);
#X obj 407 29 max 0.001;
80 #X obj 223 376 expr pow(10 \, -3*$f2/$f1);
#X obj 233 396 expr pow(10 \, -3*$f2/$f1);
#X obj 243 416 expr pow(10 \, -3*$f2/$f1);
#X obj 253 436 expr pow(10 \, -3*$f2/$f1);
#X obj 263 456 expr pow(10 \, -3*$f2/$f1);
85 #X obj 273 476 expr pow(10 \, -3*$f2/$f1);
#X obj 283 496 expr pow(10 \, -3*$f2/$f1);
#X obj 403 356 expr pow(10 \, -3*$f2/$f1);
#X obj 413 376 expr pow(10 \, -3*$f2/$f1);
#X obj 423 396 expr pow(10 \, -3*$f2/$f1);
90 #X obj 433 416 expr pow(10 \, -3*$f2/$f1);
#X obj 443 436 expr pow(10 \, -3*$f2/$f1);
#X obj 453 456 expr pow(10 \, -3*$f2/$f1);
#X obj 463 476 expr pow(10 \, -3*$f2/$f1);
#X obj 473 496 expr pow(10 \, -3*$f2/$f1);
95 #X obj 266 30 f;
#X obj 189 31 t b a;
#X text 473 527 g(i) = 10^(-3 * delay(i) / reverbTime);
#X connect 0 0 48 1;
#X connect 0 0 49 0;
100 #X connect 0 1 48 2;
#X connect 0 1 57 0;
#X connect 0 2 48 3;
#X connect 0 2 50 0;
#X connect 0 3 48 4;
105 #X connect 0 3 58 0;
#X connect 0 4 48 5;
#X connect 0 4 51 0;
#X connect 0 5 48 6;
#X connect 0 5 59 0;
110 #X connect 0 6 48 7;
#X connect 0 6 52 0;
#X connect 0 7 48 8;
#X connect 0 7 60 0;
#X connect 0 8 48 9;
115 #X connect 0 8 53 0;
#X connect 0 9 48 10;
#X connect 0 9 61 0;
#X connect 0 10 48 11;
#X connect 0 10 54 0;
120 #X connect 0 11 48 12;
#X connect 0 11 62 0;
#X connect 0 12 48 13;
#X connect 0 12 55 0;

```

```

#X connect 0 13 48 14;
125 #X connect 0 13 63 0;
#X connect 0 14 48 15;
#X connect 0 14 56 0;
#X connect 0 15 48 16;
#X connect 0 15 64 0;
130 #X connect 1 0 0 1;
#X connect 2 0 0 2;
#X connect 3 0 0 3;
#X connect 4 0 0 4;
#X connect 5 0 0 5;
135 #X connect 6 0 0 6;
#X connect 7 0 0 7;
#X connect 8 0 0 8;
#X connect 9 0 0 9;
#X connect 10 0 0 10;
140 #X connect 11 0 0 11;
#X connect 12 0 0 12;
#X connect 13 0 0 13;
#X connect 14 0 0 14;
#X connect 15 0 0 15;
145 #X connect 16 0 0 16;
#X connect 17 0 1 0;
#X connect 17 0 65 1;
#X connect 17 1 2 0;
#X connect 17 1 67 1;
150 #X connect 17 2 3 0;
#X connect 17 2 68 1;
#X connect 17 3 4 0;
#X connect 17 3 69 1;
#X connect 17 4 5 0;
155 #X connect 17 4 70 1;
#X connect 17 5 6 0;
#X connect 17 5 71 1;
#X connect 17 6 7 0;
#X connect 17 6 72 1;
160 #X connect 17 7 8 0;
#X connect 17 7 73 1;
#X connect 17 8 9 0;
#X connect 17 8 74 1;
#X connect 17 9 10 0;
165 #X connect 17 9 75 1;
#X connect 17 10 11 0;
#X connect 17 10 76 1;
#X connect 17 11 12 0;
#X connect 17 11 77 1;
170 #X connect 17 12 13 0;
#X connect 17 12 78 1;
#X connect 17 13 14 0;
#X connect 17 13 79 1;
#X connect 17 14 15 0;
175 #X connect 17 14 80 1;
#X connect 17 15 16 0;
#X connect 17 15 81 1;
#X connect 18 0 83 0;
#X connect 19 0 0 1;
180 #X connect 19 1 0 2;
```

```
#X connect 19 2 0 3;
#X connect 19 3 0 4;
#X connect 19 4 0 5;
#X connect 19 5 0 6;
185 #X connect 19 6 0 7;
#X connect 19 7 0 8;
#X connect 19 8 0 9;
#X connect 19 9 0 10;
#X connect 19 10 0 11;
190 #X connect 19 11 0 12;
#X connect 19 12 0 13;
#X connect 19 13 0 14;
#X connect 19 14 0 15;
#X connect 19 15 0 16;
195 #X connect 20 0 19 1;
#X connect 21 0 19 2;
#X connect 22 0 82 0;
#X connect 23 0 25 0;
#X connect 24 0 23 0;
200 #X connect 24 0 29 0;
#X connect 24 0 30 0;
#X connect 25 0 19 0;
#X connect 28 0 48 0;
#X connect 29 0 28 0;
205 #X connect 30 0 31 0;
#X connect 31 0 0 0;
#X connect 48 0 26 0;
#X connect 48 1 27 0;
#X connect 49 0 32 0;
210 #X connect 50 0 34 0;
#X connect 51 0 36 0;
#X connect 52 0 38 0;
#X connect 53 0 40 0;
#X connect 54 0 42 0;
215 #X connect 55 0 44 0;
#X connect 56 0 46 0;
#X connect 57 0 33 0;
#X connect 58 0 35 0;
#X connect 59 0 37 0;
220 #X connect 60 0 39 0;
#X connect 61 0 41 0;
#X connect 62 0 43 0;
#X connect 63 0 45 0;
#X connect 64 0 47 0;
225 #X connect 65 0 49 1;
#X connect 66 0 65 0;
#X connect 66 0 67 0;
#X connect 66 0 68 0;
#X connect 66 0 69 0;
230 #X connect 66 0 70 0;
#X connect 66 0 71 0;
#X connect 66 0 72 0;
#X connect 66 0 73 0;
#X connect 66 0 74 0;
235 #X connect 66 0 75 0;
#X connect 66 0 76 0;
#X connect 66 0 77 0;
```

```

#X connect 66 0 78 0;
#X connect 66 0 79 0;
240 #X connect 66 0 80 0;
#X connect 66 0 81 0;
#X connect 67 0 57 1;
#X connect 68 0 50 1;
#X connect 69 0 58 1;
245 #X connect 70 0 51 1;
#X connect 71 0 59 1;
#X connect 72 0 52 1;
#X connect 73 0 60 1;
#X connect 74 0 53 1;
250 #X connect 75 0 61 1;
#X connect 76 0 54 1;
#X connect 77 0 62 1;
#X connect 78 0 55 1;
#X connect 79 0 63 1;
255 #X connect 80 0 56 1;
#X connect 81 0 64 1;
#X connect 82 0 66 0;
#X connect 83 0 82 0;
#X connect 83 1 17 0;

```

12 keyboard-input.pd

```

#N canvas 0 0 673 355 10;
#X obj 90 108 #out window;
#X obj 90 50 loadbang;
#X msg 90 75 set_geometry 660 0 64 800 \, title h4tek-INPUT;
5 #X obj 90 138 route keypress keyrelease;
#X obj 90 172 unpack f f f s;
#X msg 171 205 key \$1 1;
#X obj 180 172 unpack f f f s;
#X msg 261 205 key \$1 0;
10 #X obj 171 277 outlet;
#X obj 420 128 ctlin;
#X obj 405 167 pack f f f;
#X msg 405 197 cc \$3 \$2 \$1;
#X connect 0 0 3 0;
15 #X connect 1 0 2 0;
#X connect 2 0 0 0;
#X connect 3 0 4 0;
#X connect 3 1 6 0;
#X connect 4 3 5 0;
20 #X connect 5 0 8 0;
#X connect 6 3 7 0;
#X connect 7 0 8 0;
#X connect 9 0 10 0;
#X connect 9 1 10 1;
25 #X connect 9 2 10 2;
#X connect 10 0 11 0;
#X connect 11 0 8 0;

```

13 keyboard-mapping.pd_lua

```

local KeyToMIDI = pd.Class:new():register("keyboard-mapping")
local notes = {

```

```

    z = 36,
    s = 37,
5    x = 38,
    d = 39,
    c = 40,
    v = 41,
    g = 42,
10   b = 43,
    h = 44,
    n = 45,
    j = 46,
    m = 47,
15
    comma = 48,
    l = 49,
    period = 50,
    semicolon = 51,
20   slash = 52,

    q = 48,
    D2 = 49,
    w = 50,
25   D3 = 51,
    e = 52,
    r = 53,
    D5 = 54,
    t = 55,
30   D6 = 56,
    y = 57,
    D7 = 58,
    u = 59,

35   i = 60,
    D9 = 61,
    o = 62,
    D0 = 63,
    p = 64,
40   bracketleft = 65,
    equal = 66,
    bracketright = 67,

    space = 0,
45   BackSpace = -1
}

function KeyToMIDI:initialize(name, atoms)
    self.inlets = 1
50   self.outlets = 2
    return true
end

function KeyToMIDI:in_1_symbol(s)
55   if (notes[s]) then
        self:outlet(1, "float", { notes[s] })
    else
        self:outlet(2, "symbol", { s })
    end
end
```

60 end

14 markiiiflex2-jack.sh

```
#!/bin/bash
pd -stderr -verbose -jack -channels 2 \
-path /usr/lib/pd/extra/lua/ -lib lua \
-path /usr/lib/pd/extra/Gem/ -lib Gem \
5 -path /usr/lib/pd/extra/zexy/ -lib zexy \
-open markiiiflex2.pd
```

15 markiiiflex2-oss.sh

```
#!/bin/bash
pd -stderr -verbose -r 48000 -oss -audiobuf 500 \
-path /usr/lib/pd/extra/lua/ -lib lua \
-path /usr/lib/pd/extra/Gem/ -lib Gem \
5 -path /usr/lib/pd/extra/zexy/ -lib zexy \
-path /usr/lib/pd/extra/iemmatrix/ -lib iemmatrix \
-open markiiiflex2.pd
```

16 markiiiflex2.pd

```
#N canvas 50 0 694 654 10;
#X obj 12 590 dac~;
#X obj 99 589 recorder~;
#X obj 165 561 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 0
5 1;
#X obj 238 531 metro 1000;
#X obj 211 576 / 1000;
#X floatatom 211 600 5 0 0 0 - - -;
#X obj 211 557 timer;
10 #X obj 186 534 t b b;
#X obj 185 508 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X msg 93 242 5 \, 6 \, 7 \, 1 \, 1 \, 1 \, 4;
#X msg 92 216 3 \, 2 \, 3 \, 2 \, 1 \, 1 \, 4;
15 #X msg 90 191 1 \, 3 \, 5 \, 3;
#X msg 66 122 5 \, 3 \, 1;
#X msg 74 147 3 \, 5 \, 1 \, 1 \, 3 \, 5;
#X msg 83 170 4 \, 6 \, 7 \, 1 \, 2 \, 3 \, 2 \, 1 \, 3 \, 5;
#X msg 95 264 6 \, 6 \, 6;
20 #X msg 92 287 7 \, 7 \, 7 \, 1;
#X floatatom 60 101 5 0 0 0 - - -;
#X obj 169 487 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1
1;
#N canvas 156 96 450 397 \ $0-audio 0;
25 #X obj 7 5 inlet;
#X obj 16 353 outlet~;
#X obj 75 354 outlet~;
#X obj 196 15 table \ $0-notes 7;
#X obj 199 84 s \ $0-notes;
30 #X msg 199 61 0 0 2 3 5 7 -4 -2;
#X obj 199 37 loadbang;
#N canvas 0 0 450 479 \ $0-organ 0;
#X obj -18 208 hip~ 5;
```

```

#X obj 181 205 hip~ 5;
35 #X obj -17 186 borgan~;
#X obj 180 181 borgan~;
#X obj -21 119 tabread \ $0-notes;
#X obj -21 165 mtof;
#X obj -19 96 - 1;
40 #X obj 185 121 tabread \ $0-notes;
#X obj 183 160 mtof;
#X obj 185 140 + 36;
#X obj 185 103 - 1;
#X obj 83 332 *~ 0.5;
45 #X obj 191 328 *~ 0.5;
#X obj 83 208 hip~ 5;
#X obj 84 182 borgan~;
#X obj 83 123 tabread \ $0-notes;
#X obj 80 161 mtof;
50 #X obj 84 100 - 1;
#X obj 82 141 + 48;
#X obj -19 145 + 36;
#X obj 311 214 hip~ 5;
#X obj 309 191 borgan~;
55 #X obj 307 121 tabread \ $0-notes;
#X obj 305 170 mtof;
#X obj 307 101 - 1;
#X obj 307 150 + 48;
#X obj 9 51 route Llo Lhi Rlo Rhi;
60 #X obj 9 71 f;
#X obj 39 71 f;
#X obj 69 71 f;
#X obj 99 71 f;
#X obj 10 8 inlet;
65 #X obj 82 360 outlet~;
#X obj 192 359 outlet~;
#X connect 0 0 11 0;
#X connect 1 0 12 0;
#X connect 2 0 0 0;
70 #X connect 3 0 1 0;
#X connect 4 0 19 0;
#X connect 5 0 2 0;
#X connect 6 0 4 0;
#X connect 7 0 9 0;
75 #X connect 8 0 3 0;
#X connect 9 0 8 0;
#X connect 10 0 7 0;
#X connect 11 0 32 0;
#X connect 12 0 33 0;
80 #X connect 13 0 11 0;
#X connect 14 0 13 0;
#X connect 15 0 18 0;
#X connect 16 0 14 0;
#X connect 17 0 15 0;
85 #X connect 18 0 16 0;
#X connect 19 0 5 0;
#X connect 20 0 12 0;
#X connect 21 0 20 0;
#X connect 22 0 25 0;
90 #X connect 23 0 21 0;

```

```

#X connect 24 0 22 0;
#X connect 25 0 23 0;
#X connect 26 0 27 0;
#X connect 26 1 28 0;
95 #X connect 26 2 29 0;
#X connect 26 3 30 0;
#X connect 27 0 6 0;
#X connect 28 0 17 0;
#X connect 29 0 10 0;
100 #X connect 30 0 24 0;
#X connect 31 0 26 0;
#X restore 11 116 pd \ $0-organ;
#X obj 9 28 route organ-note drum-hits;
#N canvas 0 0 450 435 \ $0-drums 0;
105 #X obj 28 17 inlet;
#X obj 29 385 outlet~;
#X obj 108 384 outlet~;
#X obj 91 166 snare~;
#X obj 136 167 snare~;
110 #X obj 28 48 f;
#X obj 28 76 sel 0 1 2 3 4 5 6 7;
#X msg 98 136 30;
#N canvas 0 0 450 372 \ $0-kick 0;
#X obj 37 16 inlet;
115 #X obj 40 297 outlet~;
#X obj 37 109 vline~;
#X msg 37 80 1 \, 0 166;
#X obj 36 51 t b f;
#X obj 92 114 vline~;
120 #X obj 36 139 *~;
#X obj 36 159 *~;
#X obj 36 179 *~;
#X obj 36 203 ~ 0.25;
#X obj 38 230 cos~;
125 #X obj 39 264 expr~ sin(2*tanh($v1*3));
#X connect 0 0 4 0;
#X connect 2 0 6 0;
#X connect 2 0 6 1;
#X connect 3 0 2 0;
130 #X connect 4 0 3 0;
#X connect 4 1 5 0;
#X connect 5 0 8 1;
#X connect 6 0 7 0;
#X connect 6 0 7 1;
135 #X connect 7 0 8 0;
#X connect 8 0 9 0;
#X connect 9 0 10 0;
#X connect 10 0 11 0;
#X connect 11 0 1 0;
140 #X restore 22 167 pd \ $0-kick;
#X obj 89 241 *~ 1.5;
#X obj 36 240 *~ 1.5;
#X msg 128 135 10;
#X msg 26 126 10;
145 #X msg 58 126 8;
#X obj 148 247 hverb~;
#X floatatom 195 194 5 0 0 0 - - -;

```



```
#X obj 226 250 hverb-calc 16;
#X msg 236 212 3 5 7 11;
150 #X obj 179 279 *~ 7;
#X obj 129 279 *~ 7;
#X obj 85 203 *~ 0.25;
#X obj 240 176 loadbang;
#X msg 195 171 2000;
155 #X connect 0 0 5 0;
#X connect 3 0 14 0;
#X connect 3 0 10 0;
#X connect 4 0 14 1;
#X connect 4 0 9 0;
160 #X connect 5 0 6 0;
#X connect 6 1 12 0;
#X connect 6 2 13 0;
#X connect 6 3 7 0;
#X connect 6 4 11 0;
165 #X connect 7 0 3 0;
#X connect 7 0 4 0;
#X connect 8 0 10 0;
#X connect 8 0 9 0;
#X connect 8 0 20 0;
170 #X connect 9 0 2 0;
#X connect 10 0 1 0;
#X connect 11 0 4 0;
#X connect 11 0 3 0;
#X connect 12 0 8 0;
175 #X connect 13 0 8 0;
#X connect 14 0 19 0;
#X connect 14 1 18 0;
#X connect 15 0 14 3;
#X connect 16 0 14 2;
180 #X connect 17 0 16 0;
#X connect 18 0 2 0;
#X connect 19 0 1 0;
#X connect 20 0 14 0;
#X connect 20 0 14 1;
185 #X connect 21 0 17 0;
#X connect 21 0 22 0;
#X connect 22 0 15 0;
#X restore 97 115 pd \${0}-drums;
#X obj 15 245 compress~ 48;
190 #X connect 0 0 8 0;
#X connect 5 0 4 0;
#X connect 6 0 5 0;
#X connect 7 0 10 0;
#X connect 7 1 10 1;
195 #X connect 8 0 7 0;
#X connect 8 1 9 0;
#X connect 9 0 10 0;
#X connect 9 1 10 1;
#X connect 10 0 1 0;
200 #X connect 10 1 2 0;
#X restore 19 527 pd \${0}-audio;
#N canvas 0 228 744 458 \${0}-video 0;
#X obj 46 275 outlet;
#X obj 108 275 outlet;
```

```

205  #X obj 44 14 inlet;
    #X obj 106 16 inlet;
    #X obj 205 271 outlet;
    #X obj 202 15 inlet;
    #X obj 388 13 gemhead;
210  #X obj 453 302 unpack f f f f f f;
    #X obj 453 327 pack f f f f;
    #X obj 579 403 hsv2rgb;
    #X msg 579 381 \$1 1 1;
    #X obj 453 263 route point read;
215  #X obj 385 32 t a a;
    #X msg 499 284 left \$1 \$2 \$3;
    #X obj 348 306 route read;
    #X msg 345 327 right \$1 \$2 \$3;
    #X obj 579 359 / 36;
220  #X obj 579 339 mod 36;
    #X obj 579 318 * 5;
    #X obj 416 181 t b b a b;
    #X obj 400 389 geo-pipes2 -----;
    #X obj 391 118 GEMglLineWidth 8;
225  #X obj 412 51 GEMglEnable;
    #X obj 543 16 loadbang;
    #X obj 509 52 GLdefine GLLINESMOOTH;
    #X obj 416 71 GEMglEnable;
    #X obj 512 74 GLdefine GLBLEND;
230  #X obj 410 95 GEMglShadeModel;
    #X obj 513 94 GLdefine GLSMOOTH;
    #X obj 453 353 sphere-segments2 8;
    #X obj 393 139 t a a a;
    #X msg 49 116 destroy;
235  #X obj 48 147 gemwin 25;
    #X obj 204 43 sel 0 1;
    #X msg 103 117 dimen 1576 1152 \, create :0.1 \, 1;
    #X connect 2 0 11 0;
    #X connect 3 0 14 0;
240  #X connect 5 0 33 0;
    #X connect 6 0 12 0;
    #X connect 7 0 8 0;
    #X connect 7 1 8 1;
    #X connect 7 2 8 2;
245  #X connect 7 3 8 3;
    #X connect 7 5 18 0;
    #X connect 8 0 29 0;
    #X connect 9 0 20 5;
    #X connect 10 0 9 0;
250  #X connect 11 0 7 0;
    #X connect 11 1 13 0;
    #X connect 12 0 4 0;
    #X connect 12 1 22 0;
    #X connect 13 0 29 1;
255  #X connect 14 0 15 0;
    #X connect 15 0 29 1;
    #X connect 16 0 10 0;
    #X connect 17 0 16 0;
    #X connect 18 0 17 0;
260  #X connect 19 0 0 0;
    #X connect 19 1 1 0;

```

```

#X connect 19 2 20 1;
#X connect 19 3 29 1;
#X connect 21 0 30 0;
265 #X connect 22 0 25 0;
#X connect 23 0 24 0;
#X connect 23 0 26 0;
#X connect 23 0 28 0;
#X connect 24 0 22 1;
270 #X connect 25 0 27 0;
#X connect 26 0 25 1;
#X connect 27 0 21 0;
#X connect 28 0 27 1;
#X connect 29 0 20 2;
275 #X connect 29 1 20 3;
#X connect 30 0 20 0;
#X connect 30 1 19 0;
#X connect 30 2 20 6;
#X connect 31 0 32 0;
280 #X connect 33 0 31 0;
#X connect 33 1 34 0;
#X connect 34 0 32 0;
#X restore 99 527 pd \ $0-video;
#X msg 270 440 \; pd dsp 1;
285 #N canvas 269 145 830 493 \ $0-chain 0;
#X obj 448 447 outlet;
#X obj 508 447 outlet;
#X obj 467 22 inlet;
#X obj 520 22 inlet;
290 #X obj 26 7 inlet;
#X obj 28 467 outlet;
#X obj 24 202 markiii organ-note;
#X obj 318 450 outlet;
#X msg 24 232 organ-note Llo \ $1;
295 #X msg 76 262 organ-note Llo;
#X obj 154 202 markiii organ-note;
#X obj 274 202 markiii organ-note;
#X obj 394 202 markiii organ-note;
#X msg 154 232 organ-note Lhi \ $1;
300 #X msg 206 262 organ-note Lhi;
#X msg 274 232 organ-note Rlo \ $1;
#X msg 326 262 organ-note Rlo;
#X msg 394 232 organ-note Rhi \ $1;
#X msg 446 262 organ-note Rhi;
305 #X obj 26 147 markiii organ-time;
#X obj 205 10 inlet;
#X obj 26 170 moses 0;
#X obj 83 171 t f b;
#X obj 26 106 * 111;
310 #X obj 26 128 delay;
#X msg 83 284 organ-time Llo;
#X obj 156 147 markiii organ-time;
#X obj 156 170 moses 0;
#X obj 213 171 t f b;
315 #X obj 156 106 * 111;
#X obj 156 128 delay;
#X obj 276 147 markiii organ-time;
#X obj 276 170 moses 0;

```

```
#X obj 333 171 t f b;
320 #X obj 276 106 * 111;
#X obj 276 128 delay;
#X obj 396 147 markiii organ-time;
#X obj 396 170 moses 0;
#X obj 453 171 t f b;
325 #X obj 396 106 * 111;
#X obj 396 128 delay;
#X msg 213 284 organ-time Lhi;
#X msg 333 284 organ-time Rlo;
#X msg 453 284 organ-time Rhi;
330 #X obj 205 31 route tempo;
#X obj 26 48 route Llo Lhi Rlo Rhi;
#X obj 26 82 f;
#X obj 156 82 f;
#X obj 276 86 f;
335 #X obj 396 85 f;
#X obj 205 83 * 2;
#X obj 205 50 f;
#X obj 34 408 markiii drum-hits;
#X msg 83 451 drum-hits;
340 #X msg 34 432 drum-hits \$1;
#X obj 26 27 route organ drums;
#X obj 44 304 metro 111;
#X obj 44 324 t b b;
#X obj 111 305 f;
345 #X obj 139 305 + 1;
#X obj 139 326 mod 64;
#X obj 110 344 sel 0 16 32;
#X obj 110 368 b;
#X msg 52 387 set 0 0 0;
350 #X connect 2 0 6 2;
#X connect 3 0 11 2;
#X connect 4 0 55 0;
#X connect 6 0 8 0;
#X connect 6 1 9 0;
355 #X connect 6 2 0 0;
#X connect 8 0 5 0;
#X connect 9 0 7 0;
#X connect 10 0 13 0;
#X connect 10 1 14 0;
360 #X connect 11 0 15 0;
#X connect 11 1 16 0;
#X connect 11 2 1 0;
#X connect 12 0 17 0;
#X connect 12 1 18 0;
365 #X connect 13 0 5 0;
#X connect 14 0 7 0;
#X connect 15 0 5 0;
#X connect 16 0 7 0;
#X connect 17 0 5 0;
370 #X connect 18 0 7 0;
#X connect 19 0 21 0;
#X connect 19 1 25 0;
#X connect 20 0 44 0;
#X connect 21 1 22 0;
375 #X connect 22 0 23 0;
```

```
#X connect 22 1 6 0;
#X connect 23 0 24 0;
#X connect 24 0 19 0;
#X connect 25 0 7 0;
380 #X connect 26 0 27 0;
#X connect 26 1 41 0;
#X connect 27 1 28 0;
#X connect 28 0 29 0;
#X connect 28 1 10 0;
385 #X connect 29 0 30 0;
#X connect 30 0 26 0;
#X connect 31 0 32 0;
#X connect 31 1 42 0;
#X connect 32 1 33 0;
390 #X connect 33 0 34 0;
#X connect 33 1 11 0;
#X connect 34 0 35 0;
#X connect 35 0 31 0;
#X connect 36 0 37 0;
395 #X connect 36 1 43 0;
#X connect 37 1 38 0;
#X connect 38 0 39 0;
#X connect 38 1 12 0;
#X connect 39 0 40 0;
400 #X connect 40 0 36 0;
#X connect 41 0 7 0;
#X connect 42 0 7 0;
#X connect 43 0 7 0;
#X connect 44 0 51 0;
405 #X connect 45 0 46 0;
#X connect 45 1 47 0;
#X connect 45 2 48 0;
#X connect 45 3 49 0;
#X connect 46 0 23 0;
410 #X connect 47 0 29 0;
#X connect 48 0 34 0;
#X connect 49 0 39 0;
#X connect 50 0 29 1;
#X connect 50 0 23 1;
415 #X connect 50 0 34 1;
#X connect 50 0 39 1;
#X connect 51 0 50 0;
#X connect 51 0 56 1;
#X connect 52 0 54 0;
420 #X connect 52 1 53 0;
#X connect 53 0 7 0;
#X connect 54 0 5 0;
#X connect 55 0 45 0;
#X connect 55 1 56 0;
425 #X connect 56 0 57 0;
#X connect 57 0 52 0;
#X connect 57 1 58 0;
#X connect 58 0 59 0;
#X connect 58 0 61 0;
430 #X connect 59 0 60 0;
#X connect 60 0 58 1;
#X connect 61 0 62 0;
```

```

#X connect 61 1 62 0;
#X connect 61 2 62 0;
435 #X connect 62 0 63 0;
#X connect 63 0 52 0;
#X restore 62 432 pd \${0-chain};
#X msg 62 365 organ Llo 0 \, organ Lhi 1 \, organ Rlo 2 \, organ Rhi
3;
440 #N canvas 0 0 450 300 \${0-gui 0;
#X obj 10 7 inlet;
#X obj 9 51 route Llo Lhi Rlo Rhi;
#X obj 9 71 b;
#X obj 39 71 b;
445 #X obj 69 71 b;
#X obj 99 71 b;
#X obj 101 117 bng 15 250 50 0 empty empty empty 17 7 0 10 -260097
-1 -1;
#X obj 101 101 bng 15 250 50 0 empty empty empty 17 7 0 10 -260097
450 -1 -1;
#X obj 117 101 bng 15 250 50 0 empty empty empty 17 7 0 10 -260097
-1 -1;
#X obj 117 117 bng 15 250 50 0 empty empty empty 17 7 0 10 -260097
-1 -1;
455 #X obj 133 101 bng 15 250 50 0 empty empty empty 17 7 0 10 -257985
-1 -1;
#X obj 149 101 bng 15 250 50 0 empty empty empty 17 7 0 10 -257985
-1 -1;
#X obj 133 117 bng 15 250 50 0 empty empty empty 17 7 0 10 -257985
460 -1 -1;
#X obj 149 117 bng 15 250 50 0 empty empty empty 17 7 0 10 -257985
-1 -1;
#X obj 149 51 route Llo Lhi Rlo Rhi;
#X obj 149 71 b;
465 #X obj 179 71 b;
#X obj 209 71 b;
#X obj 239 71 b;
#X obj 9 28 route organ-time organ-note drum-hits;
#X obj 294 51 b;
470 #X obj 167 111 bng 15 250 50 0 empty empty empty 17 7 0 10 -4034 -1
-1;
#X connect 0 0 19 0;
#X connect 1 0 2 0;
#X connect 1 1 3 0;
475 #X connect 1 2 4 0;
#X connect 1 3 5 0;
#X connect 2 0 6 0;
#X connect 3 0 7 0;
#X connect 4 0 9 0;
480 #X connect 5 0 8 0;
#X connect 14 0 15 0;
#X connect 14 1 16 0;
#X connect 14 2 17 0;
#X connect 14 3 18 0;
485 #X connect 15 0 12 0;
#X connect 16 0 10 0;
#X connect 17 0 13 0;
#X connect 18 0 11 0;
#X connect 19 0 1 0;

```

```

490 #X connect 19 1 14 0;
#X connect 19 2 20 0;
#X connect 20 0 21 0;
#X coords 0 -1 1 1 85 33 2 100 100;
#X restore 75 458 pd \$0-gui;
495 #X obj 226 325 markiii organ-time;
#X obj 8 328 markiii organ-note;
#X msg 277 128 4 \, 4 \, 4 \, 4;
#X msg 292 175 2 \, 2 \, 4 \, 2 \, 2 \, 4;
#X msg 283 152 3 \, 3 \, 2 \, 3 \, 3 \, 2;
500 #X msg 298 202 2 \, 4 \, 2 \, 3 \, 2 \, 3;
#X msg 316 226 8 \, 4 \, 2 \, 1 \, 1;
#X msg 324 251 8 \, 1 \, 2 \, 3 \, 2;
#X msg 327 276 1 \, 2 \, 3 \, 4 \, 6;
#X msg 331 298 1 \, 1 \, 1 \, 1;
505 #X msg 84 395 tempo 111;
#X msg 179 393 drums 1;
#X msg 182 416 drums 0;
#X obj 156 116 markiii drum-hits;
#X msg 205 29 set 0 0 0 \, 1 \, 0 \, 2 \, 0 \, 4 \, 0 \, 3 \, 4 \,
510 1 \, 3 \, 4 \, 2 \, 4 \, 0 \, 2 \, 0;
#X msg 215 66 set 0 0 0 \, 1 \, 0 \, 2 \, 0 \, 4 \, 0 \, 2 \, 4 \,
0 \, 2 \, 4 \, 0 \, 2 \, 0;
#X msg 235 84 set 0 0 0 \, 2 \, 2 \, 1 \, 2 \, 4 \, 3 \, 4 \, 3 \,
4 \, 3;
515 #X msg 177 7 set 0 0 0 \, 1 \, 0 \, 0 \, 0 \, 1 \, 0 \, 0 \, 0 \, 1
\, 0 \, 0 \, 0 \, 1 \, 0 \, 0 \, 0;
#X obj 176 331 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X msg 276 104 set 0 0 0 \, 4 \, 3 \, 2 \, 1 \, 4 \, 3 \, 2 \, 1;
520 #X msg 153 48 set 0 0 0 \, 1 \, 0 \, 0 \, 1 \, 0 \, 0 \, 1 \, 0 \,
0;
#X msg 365 126 set 0 0 0 \, 1 \, 0 \, 4 \, 0 \, 1 \, 0 \, 4 \, 0;
#X connect 2 0 1 3;
#X connect 3 0 6 1;
525 #X connect 4 0 5 0;
#X connect 6 0 4 0;
#X connect 7 0 3 0;
#X connect 7 1 6 0;
#X connect 8 0 7 0;
530 #X connect 8 0 2 0;
#X connect 9 0 26 1;
#X connect 10 0 26 1;
#X connect 11 0 26 1;
#X connect 12 0 26 1;
535 #X connect 13 0 26 1;
#X connect 14 0 26 1;
#X connect 15 0 26 1;
#X connect 16 0 26 1;
#X connect 17 0 26 1;
540 #X connect 18 0 20 2;
#X connect 19 0 1 1;
#X connect 19 0 0 0;
#X connect 19 1 1 2;
#X connect 19 1 0 1;
545 #X connect 20 0 22 2;
#X connect 20 1 22 3;

```

```

#X connect 20 2 1 0;
#X connect 22 0 19 0;
#X connect 22 1 24 0;
550 #X connect 22 2 20 0;
#X connect 22 3 20 1;
#X connect 23 0 22 0;
#X connect 27 0 25 1;
#X connect 28 0 25 1;
555 #X connect 29 0 25 1;
#X connect 30 0 25 1;
#X connect 31 0 25 1;
#X connect 32 0 25 1;
#X connect 33 0 25 1;
560 #X connect 34 0 25 1;
#X connect 35 0 22 1;
#X connect 36 0 22 0;
#X connect 37 0 22 0;
#X connect 39 0 38 1;
565 #X connect 40 0 38 1;
#X connect 41 0 38 1;
#X connect 42 0 38 1;
#X connect 43 0 36 0;
#X connect 43 0 23 0;
570 #X connect 44 0 38 1;
#X connect 45 0 38 1;
#X connect 46 0 38 1;

```

17 markiiiflex2.sh

```

#!/bin/bash
pd -stderr -verbose -alsa -audiodev 6 -channels 2 -audiobuf 500 \
-path /usr/lib/pd/extra/lua/ -lib lua \
-path /usr/lib/pd/extra/Gem/ -lib Gem \
5 -path /usr/lib/pd/extra/zexy/ -lib zexy \
-open markiiiflex2.pd

```

18 markiiiflex-jack.sh

```

#!/bin/bash
pd -stderr -verbose -jack -channels 2 \
-path /usr/lib/pd/extra/lua/ -lib lua \
-path /usr/lib/pd/extra/Gem/ -lib Gem \
5 -open markiiiflex.pd

```

19 markiiiflex-oss.sh

```

#!/bin/bash
pd -stderr -verbose -r 48000 -oss -audiobuf 500 \
-path /usr/lib/pd/extra/lua/ -lib lua \
-path /usr/lib/pd/extra/Gem/ -lib Gem \
5 -path /usr/lib/pd/extra/zexy/ -lib zexy \
-open markiiiflex.pd

```

20 markiiiflex.pd


```

#N canvas 295 40 968 655 10;
#X obj 89 88 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1 1
;
#X msg 230 55 create;
5 #X obj 212 55 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1
1;
#X msg 337 38 destroy;
#X obj 48 510 dac~;
#X msg 284 59 create :0.1;
10 #X msg 91 47 \; pd dsp 1;
#X obj 43 408 hip~ 5;
#X obj 95 411 hip~ 5;
#X obj 96 509 recorder~;
#X obj 211 440 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 0
15 1;
#X msg 243 36 dimen 1440 1080;
#X obj 289 383 metro 1000;
#X obj 264 429 / 1000;
#X floatatom 264 453 5 0 0 0 - - -;
20 #X obj 264 410 timer;
#X obj 239 358 t b b;
#X obj 238 331 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X msg 248 18 dimen 788 576;
25 #X obj 388 13 gemhead;
#X obj 404 246 t a b a a;
#X obj 453 373 t b a;
#X obj 399 394 gemlist;
#X obj 403 98 t a b;
30 #X obj 415 133 rotateXYZ;
#X obj 488 107 mod 360;
#X obj 492 88 + 1;
#X obj 453 302 unpack f f f f f f;
#X obj 453 327 pack f f f f f;
35 #X obj 580 447 hsv2rgb;
#X msg 580 425 \$1 1 1;
#X obj 415 200 scale 2.5;
#X obj 453 279 route point read;
#X obj 418 168 sphere 0.125 20;
40 #X obj 403 75 GEMglLineWidth 3;
#X obj 398 465 geo-pipe;
#X obj 398 415 color;
#X obj 450 107 f 0;
#X obj 523 403 * 0.02;
45 #X obj 406 54 t a a;
#X obj 776 127 light;
#X msg 298 86 lighting 1;
#X obj 763 101 translateXYZ 4 4 4;
#X obj 763 77 separator;
50 #X obj 591 132 light;
#X obj 578 82 separator;
#X obj 92 450 *~ 1;
#X obj 42 450 *~ 1;
#X floatatom 145 83 5 0 0 0 - - -;
55 #X obj 611 223 light;
#X obj 598 173 separator;
#X obj 578 106 translateXYZ -4 -4 4;

```

```

#X obj 598 197 translateXYZ -4 4 -4;
#X obj 791 223 light;
60 #X obj 778 173 separator;
#X obj 778 197 translateXYZ 4 4 -4;
#X obj 453 353 sphere-segments 7;
#X obj 385 32 t a a;
#X obj 230 84 gemwin 25;
65 #X obj 90 110 metro 500;
#X msg 191 377 0;
#X obj 575 329 mod 16;
#X obj 575 355 / 16;
#X obj 577 384 swap 1;
70 #X obj 577 404 -;
#X obj 170 355 spigot 0;
#X obj 216 331 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 0
1;
#X obj 44 367 borgan~;
75 #X obj 94 367 borgan~;
#X floatatom 154 299 5 0 0 0 - - -;
#X obj 48 307 tabread \$0-notes;
#X obj 32 257 table \$0-notes 7;
#X obj 46 346 mtof;
80 #X obj 36 229 s \$0-notes;
#X msg 36 183 0 0 2 3 5 7 -4 -2;
#X obj 36 159 loadbang;
#X obj 48 326 + 36;
#X obj 48 287 - 1;
85 #X msg 179 237 1 \, 1 \, 4 \, 4 \, 1 \, 1 \, 5 \, 5 \, 1 \, 1 \, 5
\, 4 \, 1 \, 1 \, 4;
#X msg 167 213 5 \, 6 \, 7 \, 1 \, 1 \, 1 \, 4;
#X msg 187 188 3 \, 2 \, 3 \, 2 \, 1 \, 1 \, 4;
#X msg 161 164 1 \, 3 \, 5 \, 3;
90 #X msg 178 143 5 \, 3 \, 1;
#X msg 193 118 3 \, 5 \, 1 \, 1 \, 3 \, 5;
#X msg 236 144 4 \, 6 \, 7 \, 1 \, 2 \, 3 \, 2 \, 1 \, 3 \, 5;
#X obj 144 272 markiii demo-6;
#X msg 10 289 0;
95 #X msg 253 164 6 \, 6 \, 6;
#X msg 321 178 7 \, 7 \, 7 \, 1;
#X connect 0 0 59 0;
#X connect 1 0 58 0;
#X connect 2 0 58 0;
100 #X connect 3 0 58 0;
#X connect 5 0 58 0;
#X connect 7 0 47 0;
#X connect 8 0 46 0;
#X connect 10 0 9 3;
105 #X connect 11 0 58 0;
#X connect 12 0 15 1;
#X connect 13 0 14 0;
#X connect 15 0 13 0;
#X connect 16 0 12 0;
110 #X connect 16 1 15 0;
#X connect 17 0 16 0;
#X connect 18 0 58 0;
#X connect 19 0 57 0;
#X connect 20 1 85 2;

```

```
115  #X connect 20 2 22 1;
      #X connect 21 0 22 0;
      #X connect 21 1 35 1;
      #X connect 22 0 36 0;
      #X connect 23 0 24 0;
120  #X connect 23 1 37 0;
      #X connect 23 1 56 1;
      #X connect 24 0 33 0;
      #X connect 25 0 37 1;
      #X connect 26 0 25 0;
125  #X connect 27 0 28 0;
      #X connect 27 1 28 1;
      #X connect 27 2 28 2;
      #X connect 27 3 28 3;
      #X connect 27 4 38 0;
130  #X connect 27 5 61 0;
      #X connect 28 0 56 0;
      #X connect 29 0 36 1;
      #X connect 30 0 29 0;
      #X connect 31 0 20 0;
135  #X connect 32 0 27 0;
      #X connect 32 1 56 1;
      #X connect 33 0 31 0;
      #X connect 34 0 23 0;
      #X connect 36 0 35 0;
140  #X connect 37 0 24 2;
      #X connect 37 0 24 1;
      #X connect 37 0 26 0;
      #X connect 38 0 35 3;
      #X connect 39 0 34 0;
145  #X connect 39 1 43 0;
      #X connect 39 1 45 0;
      #X connect 39 1 50 0;
      #X connect 39 1 54 0;
      #X connect 41 0 58 0;
150  #X connect 42 0 40 0;
      #X connect 43 0 42 0;
      #X connect 45 0 51 0;
      #X connect 46 0 4 1;
      #X connect 46 0 9 2;
155  #X connect 47 0 4 0;
      #X connect 47 0 9 1;
      #X connect 48 0 59 1;
      #X connect 50 0 52 0;
      #X connect 51 0 44 0;
160  #X connect 52 0 49 0;
      #X connect 54 0 55 0;
      #X connect 55 0 53 0;
      #X connect 56 0 21 0;
      #X connect 56 1 35 2;
165  #X connect 57 0 9 0;
      #X connect 57 1 39 0;
      #X connect 59 0 85 0;
      #X connect 60 0 10 0;
      #X connect 60 0 65 1;
170  #X connect 61 0 62 0;
      #X connect 62 0 63 0;
```

```

#X connect 63 0 64 0;
#X connect 63 1 64 1;
#X connect 64 0 30 0;
175 #X connect 65 0 60 0;
#X connect 66 0 65 1;
#X connect 67 0 7 0;
#X connect 68 0 8 0;
#X connect 70 0 76 0;
180 #X connect 72 0 67 0;
#X connect 72 0 68 0;
#X connect 74 0 73 0;
#X connect 75 0 74 0;
#X connect 76 0 72 0;
185 #X connect 77 0 70 0;
#X connect 78 0 85 1;
#X connect 79 0 85 1;
#X connect 80 0 85 1;
#X connect 81 0 85 1;
190 #X connect 82 0 85 1;
#X connect 83 0 85 1;
#X connect 84 0 85 1;
#X connect 85 0 69 0;
#X connect 85 0 77 0;
195 #X connect 85 1 65 0;
#X connect 85 2 32 0;
#X connect 86 0 77 0;
#X connect 87 0 85 1;
#X connect 88 0 85 1;

```

21 markiiiflex.sh

```

#!/bin/bash
pd -stderr -verbose -alsa -audiodev 6 -audiobuf 500 \
-path /usr/lib/pd/extra/lua/ -lib lua \
-path /usr/lib/pd/extra/Gem/ -lib Gem \
5 -open markiiiflex.pd

```

22 markiii-help-old.pd

```

#N canvas 0 0 589 582 10;
#X obj 144 205 markiii help;
#X obj 213 233 print end;
#X msg 238 166 1 \, 1 \, 2 \, 1 \, 3 \, 2 \, 2 \, 3 \, 1;
5 #X obj 144 146 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X msg 213 141 1 \, 1 \, 3 \, 2 \, 3 \, 2 \, 1 \, 1 \, 1;
#X obj 144 298 osc~;
#X obj 171 507 dac~;
10 #X obj 92 147 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1
1;
#X obj 144 276 * 440;
#X obj 44 204 markiii help;
#X obj 44 297 osc~;
15 #X obj 44 275 * 440;
#X floatatom 161 240 5 0 0 0 - - -;
#X floatatom 61 239 5 0 0 0 - - -;
#X msg 253 188 1 \, 1.5 \, 1 \, 1.5 \, 2.5 \, 1 \, 1 \, 1;

```

```

#X msg 259 214 2 \, 2 \, 2.5 \, 2 \, 1.5 \, 2.5 \, 4 \, 1;
20 #X obj 205 348 markiii help;
#X obj 245 406 osc~;
#X obj 142 380 *~ 0.5;
#X obj 243 267 f 0;
#X obj 293 267 + 1;
25 #X obj 244 377 * 110;
#X obj 245 449 clip~ -0.5 0.5;
#X obj 42 378 *~ 0.5;
#X obj 245 427 *~ 2;
#X text 35 10 markiii -- online 3d markov chain implemented in lua
30 ;
#X text 62 59 right hand inlet builds a chain;
#X text 63 77 left hand inlet generates using the chain;
#X text 62 96 left outlet is generated data;
#X text 63 114 right hand outlet bangs at logical conclusion;
35 #X text 61 39 first creation argument is name of the chain;
#X obj 92 174 metro 180;
#X msg 287 234 4 \, 3 \, 2 \, 1 \, 4 \, 3 \, 2 \, 1;
#X obj 289 347 markiii help;
#X obj 243 304 sel 0 1;
40 #X floatatom 289 378 5 0 0 0 - - -;
#X floatatom 205 375 5 0 0 0 - - -;
#X obj 297 293 mod 2;
#X msg 336 258 1.5 \, 1.5 \, 1.5 \, 1.5;
#X msg 340 282 1.5 \, 1.5 \, 1.5 \, 4;
45 #X msg 348 307 1.5 \, 1.5 \, 1.5 \, 1.5 \, 4 \, 3 \, 2 \, 1;
#X msg 348 327 1.5 \, 1.5 \, 1.5 \, 1.5 \, 2 \, 2 \, 2 \, 2;
#X obj 194 555 writesf~ 2;
#X msg 223 530 open -bytes 4 markiii-demo-1.wav \, start;
#X msg 297 495 stop;
50 #X msg 386 348 4 \, 3 \, 2 \, 1 \, 2 \, 3 \, 3.5 \, 3.5;
#X msg 388 371 3.5 \, 3.5 \, 3.75 \, 3.875 \, 4 \, 4 \, 4 \, 4;
#X connect 0 0 8 0;
#X connect 0 0 12 0;
#X connect 0 1 1 0;
55 #X connect 2 0 0 1;
#X connect 3 0 0 0;
#X connect 4 0 0 1;
#X connect 5 0 18 0;
#X connect 7 0 31 0;
60 #X connect 8 0 5 0;
#X connect 9 0 11 0;
#X connect 9 0 13 0;
#X connect 10 0 23 0;
#X connect 11 0 10 0;
65 #X connect 14 0 0 1;
#X connect 15 0 0 1;
#X connect 16 0 21 0;
#X connect 16 0 36 0;
#X connect 17 0 24 0;
70 #X connect 18 0 6 1;
#X connect 18 0 42 1;
#X connect 19 0 20 0;
#X connect 19 0 34 0;
#X connect 20 0 37 0;
75 #X connect 21 0 17 0;

```

```

#X connect 22 0 6 1;
#X connect 22 0 6 0;
#X connect 22 0 42 1;
#X connect 22 0 42 0;
80 #X connect 23 0 6 0;
#X connect 23 0 42 0;
#X connect 24 0 22 0;
#X connect 31 0 0 0;
#X connect 31 0 9 0;
85 #X connect 31 0 19 0;
#X connect 32 0 0 1;
#X connect 33 0 21 0;
#X connect 33 0 35 0;
#X connect 34 0 16 0;
90 #X connect 34 1 33 0;
#X connect 37 0 19 1;
#X connect 38 0 0 1;
#X connect 39 0 0 1;
#X connect 40 0 0 1;
95 #X connect 41 0 0 1;
#X connect 43 0 42 0;
#X connect 44 0 42 0;
#X connect 45 0 0 1;
#X connect 46 0 0 1;

```

23 markiii.pd_lua

```

local markiii = pd.Class:new():register("markiii")
local shared = { }

math.randomseed(4321234)

5
function markiii:initialize(sel, atoms)
    if type(atoms[1]) == "string" then
        if shared[atoms[1]] == nil then
            local chain = { }
10            chain[0] = { }
            chain[0][0] = { }
            chain[0][0][0] = { total = 0 }
            local visited = { }
            visited[0] = { }
15            visited[0][0] = { }
            visited[0][0][0] = { }
            shared[atoms[1]] = { chain = chain, visited = visited, count = 1 }
        else
            shared[atoms[1]].count = shared[atoms[1]].count + 1
20        end
        self.name = atoms[1]
        self.ri = 0
        self.rj = 0
        self.rk = 0
25        self.wi = 0
        self.wj = 0
        self.wk = 0
        self.inlets = 3
        self.outlets = 3
30        return true
    end
end

```

```

    else
        return false
    end
end
35
function markiii:in_1_set(atoms)
    self.ri = atoms[1]
    self.rj = atoms[2]
    self.rk = atoms[3]
40 end

function markiii:in_2_set(atoms)
    self.wi = atoms[1]
    self.wj = atoms[2]
45 self.wk = atoms[3]
end

function markiii:in_3_bang()
    self:outlet(3, "read", { self.ri, self.rj, self.rk })
50 self:outlet(3, "write", { self.wi, self.wj, self.wk })
    for i,ti in pairs(shared[self.name].chain) do
        for j,tj in pairs(ti) do
            for k,tk in pairs(tj) do
                for l,p in pairs(tk) do if l ~= "total" then
55 local v = 1
                    if nil ~= shared[self.name].visited[i] and nil ~= shared[self.name].visited[i][j]
                        and nil ~= shared[self.name].visited[i][j][k] and nil ~= shared[self.name].visited[i][j][k][l] then
                        v = shared[self.name].visited[i][j][k][l]
                    end
                    self:outlet(3, "point", { i, j, k, l, p / tk.total, v })
60 end end end end end
                end
            end
        end
    end

function markiii:in_1_bang()
    local t = shared[self.name].chain[self.ri][self.rj][self.rk]
65 if t.total == 0 then
        self:outlet(2, "bang", { })
        return
    end
    local p = math.random() * t.total
70 local q = 0
    local f = 0
    for k,v in pairs(t) do if k ~= "total" then
        q = q + v
        if p <= q then
75 f = k
            break
        end
    end
end end
if shared[self.name].visited[self.ri] == nil then shared[self.name].visited[self.ri] = { } end
80 if shared[self.name].visited[self.ri][self.rj] == nil then shared[self.name].visited[self.ri][self.rj] = { } end
if shared[self.name].visited[self.ri][self.rj][self.rk] == nil then shared[self.name].visited[self.ri][self.rj][self.rk] = { } end
if shared[self.name].visited[self.ri][self.rj][self.rk][f] == nil then shared[self.name].visited[self.ri][self.rj][self.rk][f] = 0 end
end

```

```

        ↪ self.name].visited[self.ri][self.rj][self.rk][f] = 0 end
shared[self.name].visited[self.ri][self.rj][self.rk][f] = shared[self.name].↪
        ↪ visited[self.ri][self.rj][self.rk][f] + 1
self.ri = self.rj
85 self.rj = self.rk
self.rk = f
if shared[self.name].chain[self.ri] == nil then shared[self.name].chain[self.↪
        ↪ ri] = { } end
if shared[self.name].chain[self.ri][self.rj] == nil then shared[self.name].↪
        ↪ chain[self.ri][self.rj] = { } end
if shared[self.name].chain[self.ri][self.rj][self.rk] == nil then shared[self.↪
        ↪ name].chain[self.ri][self.rj][self.rk] = { total = 0 } end
90 self:outlet(1, "float", { f })
end

function markiii:in_2_float(f)
    local n = shared[self.name].chain[self.wi][self.wj][self.wk][f]
95    local t = shared[self.name].chain[self.wi][self.wj][self.wk].total
    if n == nil then n = 0 end
    shared[self.name].chain[self.wi][self.wj][self.wk][f] = n + 1
    shared[self.name].chain[self.wi][self.wj][self.wk].total = t + 1
    self.wi = self.wj
100 self.wj = self.wk
    self.wk = f
    if shared[self.name].chain[self.wi] == nil then shared[self.name].chain[self.↪
        ↪ wi] = { } end
    if shared[self.name].chain[self.wi][self.wj] == nil then shared[self.name].↪
        ↪ chain[self.wi][self.wj] = { } end
    if shared[self.name].chain[self.wi][self.wj][self.wk] == nil then shared[self.↪
        ↪ name].chain[self.wi][self.wj][self.wk] = { total = 0 } end
105 end

function markiii:in_2_symbol(s)
    if s == "total" then self:error("'total' not allowed") else self:in_2_float(s)↪
        ↪ end
end
110

function markiii:finalize()
    shared[self.name].count = shared[self.name].count - 1
    if shared[self.name].count == 0 then
        shared[self.name] = nil
115 end
end

```

24 markiii-sonify.pd

```

#N canvas 412 263 450 388 10;
#X obj 32 25 inlet;
#X obj 51 292 outlet~;
#X obj 112 288 outlet~;
5 #X obj 131 82 t f b b;
#X obj 203 109 v \${0-j};
#X obj 205 134 v \${0-i};
#X obj 156 109 v \${0-k};
#X obj 156 136 v \${0-j};
10 #X obj 110 138 v \${0-k};
#X obj 32 48 t b f;

```



```

#X obj 33 136 t b b b;
#X obj 129 162 v \ $0-i;
#X obj 80 164 v \ $0-j;
15 #X obj 34 166 v \ $0-k;
#X obj 129 227 borgan~;
#X obj 111 261 *~ 0.5;
#X obj 51 261 *~ 0.5;
#X obj 79 227 borgan~;
20 #X obj 29 227 borgan~;
#X obj 130 192 * 40;
#X obj 79 192 * 40;
#X obj 34 193 * 40;
#X connect 0 0 9 0;
25 #X connect 3 0 8 0;
#X connect 3 1 6 0;
#X connect 3 2 4 0;
#X connect 4 0 5 0;
#X connect 6 0 7 0;
30 #X connect 9 0 10 0;
#X connect 9 1 3 0;
#X connect 10 0 13 0;
#X connect 10 1 12 0;
#X connect 10 2 11 0;
35 #X connect 11 0 19 0;
#X connect 12 0 20 0;
#X connect 13 0 21 0;
#X connect 14 0 15 0;
#X connect 15 0 2 0;
40 #X connect 16 0 1 0;
#X connect 17 0 16 0;
#X connect 17 0 15 0;
#X connect 18 0 16 0;
#X connect 19 0 14 0;
45 #X connect 20 0 17 0;
#X connect 21 0 18 0;

```

25 markiii-visualize.pd

```

#N canvas 113 60 765 541 10;
#X obj 371 483 GEMglBegin;
#X obj 428 456 GLdefine GL_LINES;
#X obj 428 434 loadbang;
5 #X obj 233 461 GEMglVertex3f;
#X obj 146 461 GEMglVertex3f;
#X obj 92 471 GEMglEnd;
#X obj 196 321 t b a;
#X obj 196 343 gemlist;
10 #X obj 191 367 t a a a;
#X obj 181 240 t a b a a;
#X obj 197 300 route point read write;
#X obj 165 121 translate -1 0 0 0;
#X obj 368 31 unpack f f f;
15 #X obj 368 52 sig~;
#X obj 428 55 sig~;
#X obj 486 57 sig~;
#X obj 369 98 snapshot~;
#X obj 429 98 snapshot~;

```

```
20  #X obj 489 98 snapshot~;
    #X obj 163 33 t a b b b;
    #X obj 370 117 pack f f f;
    #X obj 286 486 GEMglColor4f;
    #X obj 162 14 alpha;
25  #X obj 390 388 loadbang;
    #X obj 341 390 hsv2rgb;
    #X obj 332 448 unpack f f f;
    #X msg 355 369 \$1 1 1;
    #X obj 198 262 outlet;
30  #X obj 197 280 inlet;
    #X obj 355 10 inlet;
    #X obj 362 244 * \$1;
    #X obj 362 267 + \$2;
    #X obj 442 242 textfile;
35  #X msg 442 194 clear;
    #X msg 497 220 rewind;
    #X obj 451 218 until;
    #X obj 484 175 t b b a;
    #X obj 162 221 t a a b;
40  #X obj 443 264 t b a;
    #X obj 443 291 gemlist;
    #X obj 163 52 scale 0.6;
    #X obj 444 317 t a a;
    #X obj 464 153 t a a a;
45  #X obj 445 380 GEMglVertex3f;
    #X obj 462 354 GEMglColor4f;
    #X obj 486 330 unpack f f f f;
    #X obj 497 284 loadbang;
    #X obj 557 242 GEMglBegin;
50  #X obj 614 195 loadbang;
    #X obj 614 217 GLdefine GLPOINTS;
    #X obj 554 268 GEMglEnd;
    #X obj 163 139 GEMglEnable;
    #X obj 163 161 GEMglEnable;
55  #X obj 286 154 GLdefine GLBLEND;
    #X obj 285 115 loadbang;
    #X obj 286 135 GLdefine GLPOINTSMOOTH;
    #X obj 162 200 GEMglPointSize 48;
    #X obj 162 180 GEMglLineWidth 3;
60  #X msg 302 222 add \$1 \$2 \$3 \$4 \$5 \$6;
    #X obj 234 428 unpack f f f f f f;
    #X obj 361 286 +;
    #X obj 585 337 unpack f f f f f f;
    #X obj 395 285 / 60;
65  #X obj 360 305 * 360;
    #X obj 359 324 mod 360;
    #X obj 349 347 / 360;
    #X msg 494 309 1 0 0 0.2;
    #X obj 368 73 lop~ 1.5;
70  #X obj 429 76 lop~ 1.5;
    #X obj 487 78 lop~ 1.5;
    #X obj 162 77 color 1 1 1 1;
    #X obj 161 97 sphere 0.2 20;
    #X msg 399 422 1;
75  #X connect 1 0 0 1;
    #X connect 2 0 1 0;
```

```
#X connect 6 0 7 0;
#X connect 6 1 59 0;
#X connect 7 0 8 0;
80 #X connect 8 0 4 0;
#X connect 8 1 3 0;
#X connect 8 2 21 0;
#X connect 9 0 5 0;
#X connect 9 1 27 0;
85 #X connect 9 2 7 1;
#X connect 9 3 0 0;
#X connect 10 0 6 0;
#X connect 10 0 58 0;
#X connect 10 1 12 0;
90 #X connect 11 0 51 0;
#X connect 12 0 13 0;
#X connect 12 1 14 0;
#X connect 12 2 15 0;
#X connect 13 0 67 0;
95 #X connect 14 0 68 0;
#X connect 15 0 69 0;
#X connect 16 0 20 0;
#X connect 17 0 20 1;
#X connect 18 0 20 2;
100 #X connect 19 0 40 0;
#X connect 19 1 16 0;
#X connect 19 2 17 0;
#X connect 19 3 18 0;
#X connect 20 0 11 2;
105 #X connect 22 0 19 0;
#X connect 23 0 72 0;
#X connect 24 0 25 0;
#X connect 25 0 21 1;
#X connect 25 1 21 2;
110 #X connect 25 2 21 3;
#X connect 26 0 24 0;
#X connect 28 0 10 0;
#X connect 29 0 22 0;
#X connect 30 0 31 0;
115 #X connect 31 0 60 0;
#X connect 32 0 38 0;
#X connect 32 1 35 1;
#X connect 33 0 32 0;
#X connect 34 0 32 0;
120 #X connect 35 0 32 0;
#X connect 36 0 35 0;
#X connect 36 1 34 0;
#X connect 36 2 39 1;
#X connect 37 0 42 0;
125 #X connect 37 1 9 0;
#X connect 37 2 33 0;
#X connect 38 0 39 0;
#X connect 38 1 61 0;
#X connect 40 0 70 0;
130 #X connect 41 0 43 0;
#X connect 41 1 44 0;
#X connect 42 0 50 0;
#X connect 42 1 36 0;
```

```

#X connect 42 2 47 0;
135 #X connect 45 0 44 1;
#X connect 45 1 44 2;
#X connect 45 2 44 3;
#X connect 45 3 44 4;
#X connect 46 0 66 0;
140 #X connect 48 0 49 0;
#X connect 49 0 47 1;
#X connect 51 0 52 0;
#X connect 52 0 57 0;
#X connect 53 0 52 1;
145 #X connect 54 0 53 0;
#X connect 54 0 55 0;
#X connect 55 0 51 1;
#X connect 56 0 37 0;
#X connect 57 0 56 0;
150 #X connect 58 0 32 0;
#X connect 59 0 4 1;
#X connect 59 1 3 1;
#X connect 59 1 4 2;
#X connect 59 2 3 2;
155 #X connect 59 2 4 3;
#X connect 59 3 3 3;
#X connect 59 4 30 0;
#X connect 59 5 62 0;
#X connect 60 0 63 0;
160 #X connect 61 0 43 1;
#X connect 61 1 43 2;
#X connect 61 2 43 3;
#X connect 62 0 60 1;
#X connect 63 0 64 0;
165 #X connect 64 0 65 0;
#X connect 65 0 26 0;
#X connect 66 0 45 0;
#X connect 67 0 16 0;
#X connect 68 0 17 0;
170 #X connect 69 0 18 0;
#X connect 70 0 71 0;
#X connect 71 0 11 0;
#X connect 72 0 21 4;

```

26 recorder~.pd

```

#N canvas 0 0 450 300 10;
#X obj 23 12 inlet;
#X obj 133 10 inlet~;
#X obj 190 12 inlet~;
5 #X obj 244 13 inlet;
#X obj 130 264 writesf~ 2;
#X msg 158 235 stop;
#X obj 23 64 spigot;
#X obj 244 39 sel 1 0;
10 #X msg 243 62 1;
#X obj 58 119 spigot;
#X msg 108 120 0;
#X msg 272 63 1;
#X obj 52 175 spigot;

```

```

15  #X msg 108 173 0;
    #X obj 23 261 pix_write;
    #X obj 23 86 t a b;
    #X msg 53 232 auto 0;
    #X msg 165 178 open -bytes 4 rec.wav \, start;
20  #X msg 35 201 file rec/img 0 \, auto 1;
    #X connect 0 0 6 0;
    #X connect 1 0 4 0;
    #X connect 2 0 4 1;
    #X connect 3 0 7 0;
25  #X connect 5 0 4 0;
    #X connect 6 0 15 0;
    #X connect 7 0 8 0;
    #X connect 7 1 11 0;
    #X connect 8 0 6 1;
30  #X connect 8 0 9 1;
    #X connect 9 0 10 0;
    #X connect 9 0 17 0;
    #X connect 9 0 18 0;
    #X connect 10 0 9 1;
35  #X connect 11 0 12 1;
    #X connect 12 0 13 0;
    #X connect 12 0 5 0;
    #X connect 12 0 16 0;
    #X connect 13 0 12 1;
40  #X connect 13 0 6 1;
    #X connect 15 0 14 0;
    #X connect 15 1 9 0;
    #X connect 15 1 12 0;
    #X connect 16 0 14 0;
45  #X connect 17 0 4 0;
    #X connect 18 0 14 0;

```

27 rhythmiiii.pd

```

#N canvas 0 0 814 689 10;
#X obj 95 60 metro 111;
#X obj 95 81 t b b;
#X obj 282 64 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 0
5  1;
    #X obj 246 112 spigot;
    #X obj 95 36 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1 1
    ;
    #X obj 138 146 + 1;
10  #X obj 95 174 mod 16;
    #X obj 95 196 sel 0;
    #X obj 145 174 mod 4;
    #X obj 144 196 sel 0;
    #X obj 94 222 f 880;
15  #X obj 144 225 f 440;
    #X obj 94 248 t f b;
    #X obj 86 294 osc~;
    #X obj 85 320 *~;
    #X obj 120 296 vline~;
20  #X obj 74 346 dac~;
    #X obj 246 158 f 1;
    #X obj 275 158 + 1;

```

```

#X obj 230 204 f;
#X obj 232 6 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144 -1
25 -1;
#X msg 283 40 1;
#X obj 231 29 t b b;
#X msg 262 134 0;
#X obj 262 90 change;
30 #X obj 229 247 -;
#X obj 229 224 t f f;
#X msg 310 150 1;
#X obj 229 268 sel 0;
#X obj 157 367 markiii rhythmiii;
35 #X obj 256 288 moses 17;
#X obj 157 337 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X obj 110 377 delay;
#X obj 110 349 * 111;
40 #X obj 157 387 moses 0;
#X obj 212 491 *~;
#X obj 275 458 vline~;
#X obj 321 636 dac~;
#X obj 107 464 *~;
45 #X msg 207 462 \$1 \, 1 \$2;
#X obj 207 440 pack f f;
#X obj 211 389 t f f;
#X obj 250 409 * 111;
#X msg 276 435 0.3 \, 0 \$1;
50 #X obj 108 483 wrap~;
#X obj 134 438 vline~;
#X obj 289 384 print end;
#X msg 120 269 0.3 \, 0 50;
#X obj 108 561 cos~;
55 #X obj 330 411 phasor~ 110;
#X obj 109 502 hip~ 5;
#X obj 109 542 ~ 0.25;
#X obj 108 522 lop~ 1000;
#X obj 142 463 cos~;
60 #X msg 28 216 set 0 0 0;
#X obj 407 367 markiii rhythmiii;
#X obj 407 337 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X obj 360 377 delay;
65 #X obj 407 387 moses 0;
#X obj 462 491 *~;
#X obj 525 458 vline~;
#X obj 357 464 *~;
#X msg 457 462 \$1 \, 1 \$2;
70 #X obj 457 440 pack f f;
#X obj 468 387 t f f;
#X msg 526 435 0.3 \, 0 \$1;
#X obj 358 483 wrap~;
#X obj 384 438 vline~;
75 #X obj 539 384 print end;
#X obj 358 561 cos~;
#X obj 359 502 hip~ 5;
#X obj 359 542 ~ 0.25;
#X obj 358 522 lop~ 1000;

```

```

80  #X obj 392 463 cos~;
    #X obj 500 409 * 111;
    #X obj 360 349 * 111;
    #X obj 177 554 *~;
    #X msg 252 540 \$1 \, 1 \$2;
85  #X obj 252 518 pack f f;
    #X obj 178 573 wrap~;
    #X obj 254 565 vline~;
    #X obj 178 651 cos~;
    #X obj 179 592 hip~ 5;
90  #X obj 179 632 -~ 0.25;
    #X obj 178 612 lop~ 1000;
    #X obj 250 589 *~;
    #X obj 252 497 sqrt;
    #X obj 467 544 *~;
95  #X msg 542 530 \$1 \, 1 \$2;
    #X obj 542 508 pack f f;
    #X obj 468 563 wrap~;
    #X obj 544 555 vline~;
    #X obj 468 641 cos~;
100 #X obj 469 582 hip~ 5;
    #X obj 469 622 -~ 0.25;
    #X obj 468 602 lop~ 1000;
    #X obj 540 579 *~;
    #X obj 542 487 sqrt;
105 #X msg 349 229 3 \, 3 \, 2 \, 3 \, 3 \, 2 \, 3 \, 3 \, 3 \, 3 \, 2
    \, 2;
    #X msg 390 271 8 \, 7 \, 6 \, 5 \, 4 \, 3 \, 2 \, 1 \, 1 \, 1;
    #X obj 392 146 markiii melody;
    #X msg 581 78 1 \, 1 \, 5 \, 3 \, 1 \, 1 \, 5 \, 3;
110 #X msg 473 107 2.5 \, 3.5 \, 3 \, 2.5 \, 3.5 \, 3 \, 1 \, 1;
    #X obj 283 601 *~ 0.2;
    #X obj 386 596 *~ 0.2;
    #X obj 326 596 *~ 0.2;
    #X obj 207 417 + 2;
115 #X obj 457 417 + 2;
    #X obj 628 531 ggee/moog~;
    #X obj 621 401 -~ 0.5;
    #X obj 623 424 *~ 10;
    #X obj 626 447 clip~ -1 1;
120 #X obj 650 499 vline~;
    #X obj 700 383 + 1;
    #X obj 700 354 markiii melody;
    #X obj 623 378 phasor~ 110;
    #X msg 649 471 \$1 \, 220 111;
125 #X obj 695 497 sig~ 3;
    #X obj 627 561 expr~ tanh($v1*4);
    #X obj 30 193 sel 0;
    #X floatatom 36 144 5 0 0 0 - - -;
    #X msg 457 252 3 \, 3 \, 3 \, 3 \, 3 \, 3 \, 3 \, 3 \, 3 \, 3;
130 #X obj 303 658 writesf~ 2;
    #X msg 92 12 1;
    #X msg 479 60 1 \, 1 \, 1 \, 1;
    #X msg 496 80 4 \, 3 \, 2 \, 1;
    #X msg 566 59 4 \, 3 \, 2 \, 3 \, 2 \, 3 \, 2 \, 3 \, 4 \, 3 \, 2 \,
135 1;
    #X obj 171 412 bang;

```

```

#X obj 452 340 bang;
#X obj 620 351 * 55;
#X obj 390 177 * 110;
140 #X msg 491 146 2.25 \, 2.25 \, 4 \, 2 \, 2 \, 2 \, 4 \, 2;
#X msg 600 274 set 0 0 0 \, 3 \, 3 \, 2 \, 1 \, 2 \, 3 \, 4 \, 5 \,
6 \, 7 \, 8 \, 16 \, 32;
#X obj 700 404 * 330;
#X msg 396 636 stop;
145 #X msg 383 670 open -bytes 4 rhythmiii-demo-2.wav \, start;
#X msg 323 205 4 \, 4 \, 4 \, 4;
#X msg 665 103 1.5 \, 2.5 \, 5 \, 3 \, 1;
#X msg 628 124 5 \, 3 \, 2 \, 4 \, 4 \, 4;
#X msg 493 175 1.25 \, 1.5 \, 1.75 \, 4 \, 1 \, 4 \, 1 \, 1 \, 1;
150 #X msg 415 210 8 \, 8 \, 8 \, 8 \, 6 \, 6 \, 6 \, 6;
#X msg 407 198 4 \, 6 \, 6 \, 4 \, 6 \, 6;
#X msg 414 298 set 0 0 0 \, 7;
#X msg 583 207 2.25 \, 2.25 \, 4 \, 1.5 \, 1.25 \, 1;
#X obj 95 143 f 0;
155 #X msg 309 188 3 \, 3 \, 2 \, 4 \, 4 \, 4;
#X msg 401 24 1 \, 1 \, 1.5 \, 2 \, 1 \, 1 \, 1.5 \, 2;
#X msg 582 20 2 \, 3 \, 2 \, 3 \, 2 \, 3 \, 2 \, 3 \, 2 \, 1;
#X msg 371 42 1.5 \, 2 \, 1 \, 1.25 \, 1.25 \, 1.25 \, 1.5 \, 1.5 \,
1.5 \, 2 \, 1;
160 #X msg 325 248 6 \, 6 \, 4;
#X obj 28 169 mod 256;
#X msg 378 109 5 \, 3 \, 2;
#X msg 608 228 3 \, 3 \, 3 \, 2;
#X msg 232 332 5 \, 5 \, 5 \, 5 \, 5 \, 5 \, 5 \, 5 \, 5;
165 #X msg 187 310 1 \, 1 \, 1 \, 1 \, 2 \, 2 \, 3 \, 4 \, 5 \, 6 \, 7
\, 6 \, 5 \, 4 \, 3 \, 3 \, 2;
#X msg 511 330 16;
#X connect 0 0 1 0;
#X connect 0 0 114 0;
170 #X connect 0 0 100 0;
#X connect 1 0 144 0;
#X connect 1 1 3 0;
#X connect 2 0 3 1;
#X connect 2 0 24 0;
175 #X connect 3 0 17 0;
#X connect 4 0 0 0;
#X connect 4 0 31 0;
#X connect 4 0 56 0;
#X connect 4 0 135 0;
180 #X connect 5 0 144 1;
#X connect 6 0 7 0;
#X connect 7 0 10 0;
#X connect 7 1 8 0;
#X connect 8 0 9 0;
185 #X connect 9 0 11 0;
#X connect 10 0 12 0;
#X connect 11 0 12 0;
#X connect 12 0 13 0;
#X connect 12 1 47 0;
190 #X connect 14 0 16 0;
#X connect 14 0 16 1;
#X connect 15 0 14 1;
#X connect 17 0 18 0;

```



```
195  #X connect 18 0 17 1;
    #X connect 18 0 19 1;
    #X connect 19 0 26 0;
    #X connect 20 0 22 0;
    #X connect 21 0 2 0;
    #X connect 22 0 19 0;
200  #X connect 22 1 21 0;
    #X connect 23 0 17 0;
    #X connect 24 0 23 0;
    #X connect 24 0 27 0;
    #X connect 25 0 28 0;
205  #X connect 26 0 25 1;
    #X connect 26 1 25 0;
    #X connect 27 0 25 1;
    #X connect 28 1 30 0;
    #X connect 29 0 34 0;
210  #X connect 29 0 127 0;
    #X connect 29 1 46 0;
    #X connect 30 0 29 1;
    #X connect 31 0 29 0;
    #X connect 32 0 31 0;
215  #X connect 33 0 32 0;
    #X connect 34 1 33 0;
    #X connect 34 1 41 0;
    #X connect 35 0 103 0;
    #X connect 35 0 105 0;
220  #X connect 36 0 35 1;
    #X connect 38 0 44 0;
    #X connect 39 0 45 0;
    #X connect 40 0 39 0;
    #X connect 41 0 106 0;
225  #X connect 41 1 42 0;
    #X connect 42 0 40 1;
    #X connect 42 0 43 0;
    #X connect 42 0 78 1;
    #X connect 43 0 36 0;
230  #X connect 44 0 50 0;
    #X connect 45 0 38 1;
    #X connect 47 0 15 0;
    #X connect 48 0 35 0;
    #X connect 49 0 38 0;
235  #X connect 49 0 53 0;
    #X connect 49 0 61 0;
    #X connect 49 0 73 0;
    #X connect 49 0 76 0;
    #X connect 49 0 87 0;
240  #X connect 50 0 52 0;
    #X connect 51 0 48 0;
    #X connect 52 0 51 0;
    #X connect 53 0 35 0;
    #X connect 54 0 29 0;
245  #X connect 54 0 55 0;
    #X connect 54 0 100 0;
    #X connect 54 0 114 0;
    #X connect 55 0 58 0;
    #X connect 55 0 128 0;
250  #X connect 55 1 68 0;
```

```
#X connect 56 0 55 0;
#X connect 57 0 56 0;
#X connect 58 1 64 0;
#X connect 58 1 75 0;
255 #X connect 59 0 104 0;
#X connect 59 0 105 0;
#X connect 60 0 59 1;
#X connect 61 0 66 0;
#X connect 62 0 67 0;
260 #X connect 63 0 62 0;
#X connect 64 0 107 0;
#X connect 64 1 74 0;
#X connect 65 0 60 0;
#X connect 66 0 70 0;
265 #X connect 67 0 61 1;
#X connect 69 0 59 0;
#X connect 70 0 72 0;
#X connect 71 0 69 0;
#X connect 72 0 71 0;
270 #X connect 73 0 59 0;
#X connect 74 0 63 1;
#X connect 74 0 65 0;
#X connect 74 0 89 1;
#X connect 75 0 57 0;
275 #X connect 76 0 79 0;
#X connect 77 0 80 0;
#X connect 78 0 77 0;
#X connect 79 0 82 0;
#X connect 80 0 85 0;
280 #X connect 80 0 85 1;
#X connect 81 0 35 0;
#X connect 82 0 84 0;
#X connect 83 0 81 0;
#X connect 84 0 83 0;
285 #X connect 85 0 76 1;
#X connect 86 0 78 0;
#X connect 87 0 90 0;
#X connect 88 0 91 0;
#X connect 89 0 88 0;
290 #X connect 90 0 93 0;
#X connect 91 0 96 0;
#X connect 91 0 96 1;
#X connect 92 0 59 0;
#X connect 93 0 95 0;
295 #X connect 94 0 92 0;
#X connect 95 0 94 0;
#X connect 96 0 87 1;
#X connect 97 0 89 0;
#X connect 98 0 29 1;
300 #X connect 99 0 29 1;
#X connect 100 0 130 0;
#X connect 101 0 100 1;
#X connect 102 0 100 1;
#X connect 103 0 37 0;
305 #X connect 103 0 122 0;
#X connect 104 0 37 1;
#X connect 104 0 122 1;
```

```

#X connect 105 0 37 0;
#X connect 105 0 37 1;
310 #X connect 105 0 122 1;
#X connect 105 0 122 0;
#X connect 106 0 40 0;
#X connect 106 0 86 0;
#X connect 107 0 63 0;
315 #X connect 107 0 97 0;
#X connect 108 0 118 0;
#X connect 109 0 110 0;
#X connect 110 0 111 0;
#X connect 111 0 108 0;
320 #X connect 112 0 108 1;
#X connect 113 0 133 0;
#X connect 114 0 113 0;
#X connect 114 0 129 0;
#X connect 115 0 109 0;
325 #X connect 116 0 112 0;
#X connect 117 0 108 2;
#X connect 118 0 105 0;
#X connect 119 0 31 0;
#X connect 119 0 56 0;
330 #X connect 121 0 29 1;
#X connect 123 0 4 0;
#X connect 124 0 100 1;
#X connect 125 0 100 1;
#X connect 126 0 100 1;
335 #X connect 129 0 115 0;
#X connect 130 0 49 0;
#X connect 131 0 100 1;
#X connect 132 0 29 1;
#X connect 133 0 116 0;
340 #X connect 134 0 122 0;
#X connect 135 0 122 0;
#X connect 136 0 29 1;
#X connect 137 0 100 1;
#X connect 138 0 100 1;
345 #X connect 139 0 100 1;
#X connect 140 0 29 1;
#X connect 141 0 29 1;
#X connect 142 0 29 1;
#X connect 143 0 100 1;
350 #X connect 144 0 5 0;
#X connect 144 0 6 0;
#X connect 144 0 150 0;
#X connect 145 0 29 1;
#X connect 146 0 100 1;
355 #X connect 147 0 100 1;
#X connect 148 0 100 1;
#X connect 149 0 29 1;
#X connect 150 0 120 0;
#X connect 151 0 100 1;
360 #X connect 152 0 29 1;
#X connect 153 0 29 1;
#X connect 154 0 29 1;
#X connect 155 0 29 1;
```

28 snare~.pd

```

#N canvas 0 0 345 478 10;
#X obj 112 457 outlet~;
#X obj 39 8 inlet;
#X obj 146 94 noise~;
5 #X obj 34 178 osc~;
#X obj 100 300 *~;
#X obj 128 167 *~;
#X obj 113 324 *~;
#X obj 170 292 *~;
10 #X obj 14 100 *~;
#X obj 68 322 *~;
#X obj 14 125 *~ 12;
#X obj 107 354 clip~ 0 1;
#X obj 34 201 clip~ -0.8 0.8;
15 #X obj 166 316 *~ 0.1;
#X obj 108 375 *~ 0.4;
#X obj 54 359 *~ 0.9;
#X obj 146 118 bp~ 3300 0.5;
#X obj 146 142 lop~ 7000;
20 #X obj 112 428 bp~ 8000 1;
#X obj 73 263 clip~ -0.9 0.9;
#X obj 71 241 *~ 10000;
#X obj 15 154 +~ 150;
#X obj 113 399 *~ 0.35;
25 #X obj 40 70 cm-ead~ 1 300;
#X obj 38 49 t b f;
#X obj 116 50 * 10;
#X obj 37 29 sel 0;
#X connect 1 0 26 0;
30 #X connect 2 0 16 0;
#X connect 3 0 12 0;
#X connect 4 0 6 0;
#X connect 5 0 6 1;
#X connect 6 0 11 0;
35 #X connect 7 0 13 0;
#X connect 8 0 9 1;
#X connect 8 0 10 0;
#X connect 9 0 15 0;
#X connect 10 0 21 0;
40 #X connect 11 0 14 0;
#X connect 12 0 9 0;
#X connect 12 0 20 0;
#X connect 13 0 22 0;
#X connect 14 0 22 0;
45 #X connect 15 0 22 0;
#X connect 16 0 7 0;
#X connect 16 0 17 0;
#X connect 17 0 5 1;
#X connect 18 0 0 0;
50 #X connect 19 0 4 0;
#X connect 20 0 19 0;
#X connect 21 0 3 0;
#X connect 22 0 18 0;
#X connect 23 0 4 1;
55 #X connect 23 0 5 0;

```

```

#X connect 23 0 8 0;
#X connect 23 0 8 1;
#X connect 23 0 7 1;
#X connect 24 0 23 0;
60 #X connect 24 1 25 0;
#X connect 25 0 23 2;
#X connect 26 1 24 0;

```

29 sphere-segments2.pd_lua

```

local sphere = pd.Class:new():register("sphere-segments2")

local function mmul(l, r)
    local v = { }
    5   for i = 0,3 do
        local f = 0.0;
        for j = 0,3 do
            f = f + l[i*4+j] * r[j+1];
        end
    10    v[i+1] = f;
    end
    return v
end

15 local function mmulm(l, r)
    local m = { }
    for i = 0,3 do
        for j = 0,3 do
            local f = 0.0;
            20    for k = 0,3 do
                f = f + l[i*4+k] * r[k*4+j];
            end
            m[i*4+j] = f
        end
    25    end
    return m
end

-- /* quaternion norm squared */
30 local function qnorm2(p)
    return p[0]*p[0]+p[1]*p[1] + p[2]*p[2] + p[3]*p[3];
end

-- /* quaternion norm */
35 local function qnorm(p)
    return math.sqrt(qnorm2(p));
end

-- /* quaternion inverse: r := p^-1 */
40 -- /* p^-1 = p* / |p|^2 */
local function qinv(p)
    local pn2 = qnorm2(p);
    if pn2 == 0 then pn2 = 1 end
    local r = { }
    45    r[0] = p[0] / pn2;
    r[1] = -p[1] / pn2;
    r[2] = -p[2] / pn2;

```

```

    r[3] = -p[3] / pn2;
    return r;
50 end

local function qadd(p, q)
    local r = { }
    r[0] = p[0] + q[0]
55   r[1] = p[1] + q[1]
    r[2] = p[2] + q[2]
    r[3] = p[3] + q[3]
    return r
end
60

-- /* quaternion multiplication: r := q * p */
local function qmul(q, p)
    local q1, q2, q3, q4, p1, p2, p3, p4, r1, r2, r3, r4;
    q1 = q[0]; q2 = q[1]; q3 = q[2]; q4 = q[3];
65   p1 = p[0]; p2 = p[1]; p3 = p[2]; p4 = p[3];
    r1 = q1 * p1 - q2 * p2 - q3 * p3 - q4 * p4;
    r2 = q2 * p1 + q1 * p2 - q4 * p3 + q3 * p4;
    r3 = q3 * p1 + q4 * p2 + q1 * p3 - q2 * p4;
    r4 = q4 * p1 - q3 * p2 + q2 * p3 + q1 * p4;
70   local r = { }
    r[0] = r1; r[1] = r2; r[2] = r3; r[3] = r4;
    return r
end

75 local function qmuls(p, s)
    local r = { }
    r[0] = p[0] * s
    r[1] = p[1] * s
    r[2] = p[2] * s
80   r[3] = p[3] * s
    return r
end

local function qnormalize(p)
85   local n = qnorm(p)
    if n ~= 0 then
        return qmuls(p, 1/n)
    else
        return qmuls(p, 0)
90   end
end

-- /* quaternion exponentiation: r := exp(p) */
-- /* exp(s;v) = exp(s) (cos(|v|) ; v sin(|v|) / |v|) */
95 local function qexp(p)
    local s, v0, v1, v2, vn, se, vns, vnc;
    s = p[0];
    v0 = p[1];
    v1 = p[2];
100   v2 = p[3];
    vn = math.sqrt(v0*v0 + v1*v1 + v2*v2);
    vnc = math.cos(vn);
    if vn ~= 0 then
        vns = math.sin(vn) / vn;

```

```

105     else
        vns = 0
    end
    se = math.exp(s);
    local r = { }
110    r[0] = se * vnc;
    r[1] = v0 * vns;
    r[2] = v1 * vns;
    r[3] = v2 * vns;
    return r
115 end

-- /* quaternion logarithm: r := log(p) */
-- /* ln(q@(s;v)) = (ln(|q|) ; v acos(s / |q|) / |v|) */
local function qlog(p)
120    local s, v0, v1, v2, vn2, vn, qn, qnl, sac;
    s = p[0];
    v0 = p[1];
    v1 = p[2];
    v2 = p[3];
125    vn2 = v0*v0 + v1*v1 + v2*v2;
    vn = math.sqrt(vn2);
    qn = math.sqrt(s*s + vn2);
    qnl = math.log(qn);
    sac = math.acos(s / qn) / vn;
130    local r = { }
    r[0] = qnl;
    r[1] = v0 * sac;
    r[2] = v1 * sac;
    r[3] = v2 * sac;
135    return r;
end

-- /* quaternion scalar power: r := p^f */
-- /* p^f = exp(ln(p) f) */
140 local function qpows(p, f)
    local l = qlog(p);
    l[0] = l[0] * f;
    l[1] = l[1] * f;
    l[2] = l[2] * f;
145    l[3] = l[3] * f;
    return qexp(l);
end

-- /* quaternion slerp: r := Slerp(p,q,t) */
150 -- /* Slerp(p,q;t) = (q p^-1)^t p */
local function qslerp(p, q, t)
    return qmul(qpows(qmul(q, qinv(p)), t), p)
end

155 -- /* 4x4 rotation matrix from two axes and an angle */
local function mrot(i, j, t)
    local m = { }
    local k = 0
    for ii = 0,3 do for jj = 0,3 do
160        if (ii == i and jj == j) or (ii == j and jj == i) then
            m[k] = math.cos(t)
        end
    end
end

```

```

elseif (ii == i and jj == j) then
    m[k] = -math.sin(t)
elseif (ii == j and jj == i) then
165     m[k] = math.sin(t)
elseif (ii == jj) then
    m[k] = 1
else
    m[k] = 0
170 end
    k = k + 1
end end
return m
end
175
--[[
The cross product is defined by the formula[2]


$$\mathbf{a} \times \mathbf{b} = a b \sin \theta \ \mathbf{\hat{n}}$$

180
where theta is the measure of the smaller angle between a and b ( $0^\circ \leq \theta \leq 180^\circ$ ),
a and b are the magnitudes of vectors a and b, and  $\mathbf{\hat{n}}$ 
is a unit vector perpendicular to the plane containing a and b in the direction
given by the right-hand rule as illustrated.
185 
$$\mathbf{i}a_2b_3 + \mathbf{j}a_3b_1 + \mathbf{k}a_1b_2 - \mathbf{i}a_3b_2 - \mathbf{j}a_1b_3 - \mathbf{k}a_2b_1.$$


--]]

190 local function cross(a, b)
    return { a[2]*b[3]-a[3]*b[2], a[3]*b[1]-a[1]*b[3], a[1]*b[2] - a[2]*b[1] }
end

--[[
195 http://www.gamedev.net/community/forums/topic.asp?topic\_id=429507&whichpage=1&#2856228

@The OP: There's actually a very nice algorithm for finding the quaternion that
rotates one vector onto another that has a couple of advantages over the
200 'standard' method, namely that it works with vectors of arbitrary length,
and that it handles uniformly (and robustly) the case where the vectors are
aligned and nearly parallel.

Here's the algorithm in pseudocode:
205
quaternion q;
vector3 c = cross(v1,v2);
q.v = c;
if ( vectors are known to be unit length ) {
210     q.w = 1 + dot(v1,v2);
} else {
    q.w = sqrt(v1.length_squared() * v2.length_squared()) + dot(v1,v2);
}
q.normalize();
215 return q;

```



```

--]]

local function dot(u, v)
220   return u[1]*v[1]+u[2]*v[2]+u[3]*v[3]
end

local function qvecs3(u, v)
  local q = cross(u, v)
225   q[0] = math.sqrt(dot(u,u)*dot(v,v)) + dot(u, v)
  local r = qnorm(q)
  q[0] = q[0] / r
  q[1] = q[1] / r
  q[2] = q[2] / r
230   q[3] = q[3] / r
  return q
end

local function qvecs3short(u, v)
235   if dot(u,v) < 0 then
    return qvecs3(u, { -v[1], -v[2], -v[3] })
  else
    return qvecs3(u, v)
  end
240 end

--[[

In the same way the hyperspherical space of 3D rotations can be
245 parameterized by three angles (Euler angles), but any such parameterization
is degenerate at some points on the hypersphere, leading to the problem of
gimbal lock. We can avoid this by using four Euclidean coordinates w,x,y,z,
with  $w^2 + x^2 + y^2 + z^2 = 1$ . The point (w,x,y,z) represents a rotation around
the axis directed by the vector (x,y,z) by an angle
250  $\alpha = 2 \arccos\{w\}$   $w = 2 \arcsin\{z / \sqrt{x^2+y^2+z^2}\}$ .

--]]

-- /* 4x4 rotation matrix from one unit quaternion */
255 -- http://en.wikipedia.org/wiki/Quaternions\_and\_spatial\_rotation#%E2%99%A5\_From\_a\_quaternion\_to\_an\_orthogonal\_matrix
local function mquat(q)
  local m = { }
  local a, b, c, d
  a = q[0]; b = q[1]; c = q[2]; d = q[3]
260   m[ 0] = a*a+b*b-c*c-d*d
  m[ 1] = 2*b*c-2*a*d
  m[ 2] = 2*b*d+2*a*c
  m[ 3] = 0
  m[ 4] = 2*b*c+2*a*d
265   m[ 5] = a*a-b*b+c*c-d*d
  m[ 6] = 2*c*d-2*a*b
  m[ 7] = 0
  m[ 8] = 2*b*d-2*a*c
  m[ 9] = 2*c*d+2*a*b
270   m[10] = a*a-b*b-c*c+d*d
  m[11] = 0
  m[12] = 0

```

```

    m[13] = 0
    m[14] = 0
275   m[15] = 1
    return m
end

-- /* 4x4 rotation matrix from two unit(?) quaternions */
280 local function mquat2(q, p)
    local m = { }
    local q0, q1, q2, q3, p0, p1, p2, p3;
    q0 = q[0]; q1 = q[1]; q2 = q[2]; q3 = q[3];
    p0 = p[0]; p1 = p[1]; p2 = p[2]; p3 = p[3];
285   m[ 0] = q0*p0 + q1*p1 + q2*p2 + q3*p3;
    m[ 1] = q1*p0 - q0*p1 - q3*p2 + q2*p3;
    m[ 2] = q2*p0 + q3*p1 - q0*p2 - q1*p3;
    m[ 3] = q3*p0 - q2*p1 + q1*p2 - q0*p3;
    m[ 4] = - q1*p0 + q0*p1 - q3*p2 + q2*p3;
290   m[ 5] = q0*p0 + q1*p1 - q2*p2 - q3*p3;
    m[ 6] = - q3*p0 + q2*p1 + q1*p2 - q0*p3;
    m[ 7] = q2*p0 + q3*p1 + q0*p2 + q1*p3;
    m[ 8] = - q2*p0 + q3*p1 + q0*p2 - q1*p3;
    m[ 9] = q3*p0 + q2*p1 + q1*p2 + q0*p3;
295   m[10] = q0*p0 - q1*p1 + q2*p2 - q3*p3;
    m[11] = - q1*p0 - q0*p1 + q3*p2 + q2*p3;
    m[12] = - q3*p0 - q2*p1 + q1*p2 + q0*p3;
    m[13] = - q2*p0 + q3*p1 - q0*p2 + q1*p3;
    m[14] = q1*p0 + q0*p1 + q3*p2 + q2*p3;
300   m[15] = q0*p0 - q1*p1 - q2*p2 + q3*p3;
    return m
end

local function mrotalign(v)
305   return mquat(qvecs3(v, { math.sqrt(dot(v,v)), 0, 0 })))
end

local function radius(w)
    return math.sqrt(w[1]*w[1]+w[2]*w[2]+w[3]*w[3]+w[4]*w[4])
310 end

local function normalize(w)
    local r = radius(w)
    if r ~= 0 then
315       return { w[1] / r, w[2] / r, w[3] / r, w[4] / r }
    else
        return { 0, 0, 0, 0 }
    end
end

320 end

local function project(w)
    local u = normalize(w)
    local k = 1 / (1.01 - u[4])
    local v = { u[1] * k, u[2] * k, u[3] * k }
325   return v
end

local function pol2car(a)
    return { math.cos(a[1]), math.sin(a[1]) * math.cos(a[2]), math.sin(a[1]) *

```

```

        ↪ math.sin(a[2]) * math.cos(a[3]), math.sin(a[1]) * math.sin(a[2]) * math.↪
        ↪ sin(a[3]) }
330 end

local function car2pol(p)
end

335 local function vadd(p, q)
    return { p[1]+q[1], p[2]+q[2], p[3]+q[3], p[4]+q[4] }
end

local function vsub(p, q)
340 return { p[1]-q[1], p[2]-q[2], p[3]-q[3], p[4]-q[4] }
end

local range = 7
local function note2pol(x, y, z)
345 return { math.pi * (x%range) / range, math.pi * (y%range) / range, 2 * math.pi↪
        ↪ * (z%range) / range }
end

local function delta(p, q, n)
    return { (q[1] - p[1]) / n, (q[2] - p[2]) / n, (q[3] - p[3]) / n, (q[4] - p↪
        ↪ [4]) / n }
350 end

function sphere:initialize(sel, atoms)
    if type(atoms[1]) == "number" then
        if atoms[1] >= 1 then
355 local r2 = math.sqrt(0.5)
            self.quality = math.floor(atoms[1])
            self.inlets = 2
            self.outlets = 2
            self.cursor = { { 0, 0, 0 }, { 0, 0, 0 } }
360 self.source = { { 0, 0, 0, 0 }, { 0, 0, 0, 0 } }
            self.position = { { r2, 0, 0, r2 }, { -r2, 0, 0, r2 } }
            self.target = { { 0, 0, 0, 0 }, { 0, 0, 0, 0 } }
            self.matrix = {0,0,0, 0,1,0,0, 0,0,1,0, 0,0,0,1}
            self.matrix[0] = 1
365 self:in_2_left({ 1, 0, 0 })
            self:in_2_left({ 1, 0, 0 })
            self:in_2_right({ 5, 0, 0 })
            self:in_2_right({ 5, 0, 0 })
            return true
370 end
        end
        return false
    end

375 function sphere:in_2_left(atoms)
    if atoms[1] ~= self.cursor[1][1] or atoms[2] ~= self.cursor[1][2] or atoms[3] ↪
        ↪ ~= self.cursor[1][3] then
        self.source[1] = pol2car(note2pol(self.cursor[1][1], self.cursor[1][2], self↪
        ↪ .cursor[1][3]))
        self.cursor[1] = { atoms[1], atoms[2], atoms[3] }
    end
380 self.target[1] = pol2car(note2pol(atoms[1], atoms[2], atoms[3]))

```

```

end

function sphere:in_2_right(atoms)
  if atoms[1] ~= self.cursor[2][1] or atoms[2] ~= self.cursor[2][2] or atoms[3] ↯
    ↵ ~= self.cursor[2][3] then
385     self.source[2] = pol2car(note2pol(self.cursor[2][1], self.cursor[2][2], self ↯
        ↵ .cursor[2][3]))
    self.cursor[2] = { atoms[1], atoms[2], atoms[3] }
  end
  self.target[2] = pol2car(note2pol(atoms[1], atoms[2], atoms[3]))
end

390 local function vtoq(v)
  local q = { }
  q[0] = v[1]
  q[1] = v[2]
395  q[2] = v[3]
  q[3] = v[4]
  return q
end

400 local function mprint(s, v)
  print(s, v[0], v[1], v[2], v[3])
  print(s, v[4], v[5], v[6], v[7])
  print(s, v[8], v[9], v[10], v[11])
  print(s, v[12], v[13], v[14], v[15])
405 end
local function vprint(s, v)
  print(s, v[1], v[2], v[3], v[4])
end
local function qprint(s, v)
410  print(s, v[0], v[1], v[2], v[3])
end

function sphere:in_2_bang()

415  -- get starting positions
  local p1 = mmul(self.matrix, self.position[1])
  local q1 = mmul(self.matrix, self.position[2])
  local m1 = normalize(vadd(p1, q1))
420  -- if m1[4] < 0 then m1 = vsub({0,0,0,0}, m1) end

  -- vprint("p1", p1)
  -- vprint("q1", q1)
  -- vprint("m1", m1)
425  -- update positions
  local k1 = 0.8
  local k2 = 1 - k1
  for i = 1,2 do
430    local s = self.source[i]
    local x = self.position[i]
    local y = self.target[i]
    local z = { x[1] * k1 + k2 * y[1], x[2] * k1 + k2 * y[2], x[3] * k1 + k2 * y ↯
        ↵ [3], x[4] * k1 + k2 * y[4] }
    self.position[i] = normalize(z)
  end
end

```

```

435     end

    -- get new positions
    local p2 = mmul(self.matrix, self.position[1])
    local q2 = mmul(self.matrix, self.position[2])
440    local m2 = normalize(vadd(p2, q2))
    -- if m2[4] < 0 then m2 = vsub({0,0,0,0}, m2) end

    -- vprint("p2", p2)
    -- vprint("q2", q2)
445    -- vprint("m2", m2)

    -- find shortest rotation of m2 -> m1
    -- FIXME TODO understand why f1 is needed at all
    local r1 = mquat2(qinv(vtoq(m1)), qinv(vtoq(m2)))
450    local f1 = {0,0,0, 0,-1,0,0, 0,0,-1,0, 0,0,0,1}
    f1[0] = 1
    r1 = mmulm(f1, r1)
    -- mprint("r1", r1)

455    -- perform rotate m2 -> m1
    local p3 = mmul(r1, p2)
    local q3 = mmul(r1, q2)
    local m3 = mmul(r1, m2)

460    -- vprint("p3", p3)
    -- vprint("q3", q3)
    -- vprint("m3", m3)

    -- find shortest rotation of p3 q3 -> (+/- x4, 0, 0, w4)
465    local r2 = mrotalign(p3)

    -- mprint("r2", r2)

    -- build resulting matrix
470    local r3 = mmulm(r2, r1)
    -- mprint("r3", r3)
    local r4 = mmulm(r3, self.matrix)
    -- mprint("r4", r4)

475    self.matrix = r4

end

function sphere:in_1_list(atoms)
480    local x = atoms[1]
    local y = atoms[2]
    local z = atoms[3]
    local w = atoms[4]
    if type(x) == "number" and type(y) == "number" and type(z) == "number" and
        ↵ type(w) == "number" then
485        local n = self.quality
        local a = note2pol(x, y, z)
        local b = note2pol(y, z, w)
        local p = pol2car(a)
        local q = pol2car(b)
490        local d = delta(p, q, n)

```

```

        local s = mmul(self.matrix, p)
        --s          = mmul(self.matrix2, s)
        --  s          = mmul(self.matrix3, s)
        local t
495    local ss = project(s)
        for i = 1,n do
            t = mmul(self.matrix, { p[1] + d[1] * i, p[2] + d[2] * i, p[3] + d[3] * i,
                ↪ p[4] + d[4] * i })
            --  t = mmul(self.matrix2, t)
            --  t = mmul(self.matrix3, t)
500    local tt = project(t)
            if ss ~= nil and tt ~= nil then
                self:outlet(2, "list", ss)
                self:outlet(1, "list", tt)
            end
505    s = t
        ss = tt
        end
    end
end

```

30 sphere-segments.pd_lua

```

local sphere = pd.Class:new():register("sphere-segments")

local function mmul(l, r)
    local v = { }
    5    for i = 0,3 do
        local f = 0.0;
        for j = 0,3 do
            f = f + l[i*4+j] * r[j+1];
        end
    10    v[i+1] = f;
    end
    return v
end

15  -- /* quaternion norm squared */
    local function qnorm2(p)
        return p[0]*p[0]+p[1]*p[1] + p[2]*p[2] + p[3]*p[3];
    end

20  -- /* quaternion norm */
    local function qnorm(p)
        return math.sqrt(qnorm2(p));
    end

25  -- /* quaternion inverse: r := p^-1 */
    -- /* p^-1 = p* / |p|^2 */
    local function qinv(p)
        local pn2 = qnorm2(p);
        local r = { }
    30    r[0] = p[0] / pn2;
        r[1] = -p[1] / pn2;
        r[2] = -p[2] / pn2;
        r[3] = -p[3] / pn2;
        return r;
    end

```

```

35  end

-- /* quaternion multiplication: r := q * p */
local function qmul(q, p)
  local q1, q2, q3, q4, p1, p2, p3, p4, r1, r2, r3, r4;
40  q1 = q[0]; q2 = q[1]; q3 = q[2]; q4 = q[3];
  p1 = p[0]; p2 = p[1]; p3 = p[2]; p4 = p[3];
  r1 = q1 * p1 - q2 * p2 - q3 * p3 - q4 * p4;
  r2 = q2 * p1 + q1 * p2 - q4 * p3 + q3 * p4;
  r3 = q3 * p1 + q4 * p2 + q1 * p3 - q2 * p4;
45  r4 = q4 * p1 - q3 * p2 + q2 * p3 + q1 * p4;
  local r = { }
  r[0] = r1; r[1] = r2; r[2] = r3; r[3] = r4;
  return r
end

50
-- /* quaternion exponentiation: r := exp(p) */
-- /* exp(s;v) = exp(s) (cos(|v|) ; v sin(|v|) / |v|) */
local function qexp(p)
  local s, v0, v1, v2, vn, se, vns, vnc;
55  s = p[0];
  v0 = p[1];
  v1 = p[2];
  v2 = p[3];
  vn = math.sqrt(v0*v0 + v1*v1 + v2*v2);
60  vnc = math.cos(vn);
  vns = math.sin(vn) / vn;
  se = math.exp(s);
  local r = { }
  r[0] = se * vnc;
65  r[1] = v0 * vns;
  r[2] = v1 * vns;
  r[3] = v2 * vns;
  return r
end

70
-- /* quaternion logarithm: r := log(p) */
-- /* ln(q@(s;v)) = (ln(|q|) ; v acos(s / |q|) / |v|) */
local function qlog(p)
  local s, v0, v1, v2, vn2, vn, qn, qnl, sac;
75  s = p[0];
  v0 = p[1];
  v1 = p[2];
  v2 = p[3];
  vn2 = v0*v0 + v1*v1 + v2*v2;
80  vn = math.sqrt(vn2);
  qn = math.sqrt(s*s + vn2);
  qnl = math.log(qn);
  sac = math.acos(s / qn) / vn;
  local r = { }
85  r[0] = qnl;
  r[1] = v0 * sac;
  r[2] = v1 * sac;
  r[3] = v2 * sac;
  return r;
90  end

```

```

-- /* quaternion scalar power: r := p^f */
-- /* p^f = exp(ln(p) f) */
local function qpows(p, f)
95   local l = qlog(p);
      l[0] = l[0] * f;
      l[1] = l[1] * f;
      l[2] = l[2] * f;
      l[3] = l[3] * f;
100   return qexp(l);
end

-- /* quaternion slerp: r := Slerp(p,q,t) */
-- /* Slerp(p,q;t) = (q p^-1)^t p */
105 local function qslerp(p, q, t)
      return qmul(qpows(qmul(q, qinv(p)), t), p)
end

-- /* 4x4 rotation matrix from two unit(?) quaternions */
110 local function mquat2(q, p)
      local m = { }
      local q0, q1, q2, q3, p0, p1, p2, p3;
      q0 = q[0]; q1 = q[1]; q2 = q[2]; q3 = q[3];
      p0 = p[0]; p1 = p[1]; p2 = p[2]; p3 = p[3];
115 m[ 0] = q0*p0 + q1*p1 + q2*p2 + q3*p3;
      m[ 1] = q1*p0 - q0*p1 - q3*p2 + q2*p3;
      m[ 2] = q2*p0 + q3*p1 - q0*p2 - q1*p3;
      m[ 3] = q3*p0 - q2*p1 + q1*p2 - q0*p3;
      m[ 4] = - q1*p0 + q0*p1 - q3*p2 + q2*p3;
120 m[ 5] = q0*p0 + q1*p1 - q2*p2 - q3*p3;
      m[ 6] = - q3*p0 + q2*p1 + q1*p2 - q0*p3;
      m[ 7] = q2*p0 + q3*p1 + q0*p2 + q1*p3;
      m[ 8] = - q2*p0 + q3*p1 + q0*p2 - q1*p3;
      m[ 9] = q3*p0 + q2*p1 + q1*p2 + q0*p3;
125 m[10] = q0*p0 - q1*p1 + q2*p2 - q3*p3;
      m[11] = - q1*p0 - q0*p1 + q3*p2 + q2*p3;
      m[12] = - q3*p0 - q2*p1 + q1*p2 + q0*p3;
      m[13] = - q2*p0 + q3*p1 - q0*p2 + q1*p3;
      m[14] = q1*p0 + q0*p1 + q3*p2 + q2*p3;
130 m[15] = q0*p0 - q1*p1 - q2*p2 + q3*p3;
      return m
end

local function radius(w)
135   return math.sqrt(w[1]*w[1]+w[2]*w[2]+w[3]*w[3]+w[4]*w[4])
end

local function normalize(w)
      local r = radius(w)
140   return { w[1] / r, w[2] / r, w[3] / r, w[4] / r }
end

local function project(w)
      local u = normalize(w)
145   local k = 2 / (2 - u[4])
      local v = { u[1] * k, u[2] * k, u[3] * k }
      return v
end

```



```

150  local function pol2car(a)
      return { math.cos(a[1]), math.sin(a[1]) * math.cos(a[2]), math.sin(a[1]) * ↵
            ↵ math.sin(a[2]) * math.cos(a[3]), math.sin(a[1]) * math.sin(a[2]) * math.↵
            ↵ sin(a[3]) }
    end

    local function car2pol(p)
155  end

    local function vadd(p, q)
      return { p[1]+q[1], p[2]+q[2], p[3]+q[3], p[4]+q[4] }
    end
160
    local function vsub(p, q)
      return { p[1]-q[1], p[2]-q[2], p[3]-q[3], p[4]-q[4] }
    end

165  local range = 7
    local function note2pol(x, y, z)
      return { math.pi * (x%range) / range, math.pi * (y%range) / range, 2 * math.pi ↵
            ↵ * (z%range) / range }
    end

170  local function delta(p, q, n)
      return { (q[1] - p[1]) / n, (q[2] - p[2]) / n, (q[3] - p[3]) / n, (q[4] - p[4] ↵
            ↵ [4]) / n }
    end

    function sphere:initialize(sel, atoms)
175    if type(atoms[1]) == "number" then
        if atoms[1] >= 1 then
            self.quality = math.floor(atoms[1])
            self.inlets = 2
            self.outlets = 2
180            self.cursor = { 0, 0, 0 }
            self.source = pol2car(note2pol(0,0,0))
            self.angles = { 0, 0, 0 }
            self.position = { 0, 0, 0, 0 }
            self:in_2_list({ 0, 0, 0 })
            self:in_2_list({ 0, 0, 0 })
185            self.matrix1 = {0,0,0, 0,1,0,0, 0,0,1,0, 0,0,0,1}
            self.matrix1[0] = 1
            self.matrix2 = {0,0,0, 0,1,0,0, 0,0,1,0, 0,0,0,1}
            self.matrix2[0] = 1
            self.matrix3 = {0,0,0, 0,1,0,0, 0,0,1,0, 0,0,0,1}
190            self.matrix3[0] = 1
            return true
        end
    end
    return false
195  end

    function sphere:in_2_list(atoms)
    -- self.source = self.target
200  -- self.position = self.source
        if atoms[1] ~= self.cursor[1] or atoms[2] ~= self.cursor[2] or atoms[3] ~= ↵

```

```

        ↪ self.cursor[3] then
        self.source = pol2car(note2pol(self.cursor[1], self.cursor[2], self.cursor↵
        ↪ [3]))
        self.cursor = { atoms[1], atoms[2], atoms[3] }
    end
205     self.target = pol2car(note2pol(atoms[1], atoms[2], atoms[3]))
end

function sphere:in_2_bang()
    local s = self.source
210     local x = self.position
    local y = self.target
    local k1 = 0.7
    local k2 = 1 - k1
    local z = { x[1] * k1 + k2 * y[1], x[2] * k1 + k2 * y[2], x[3] * k1 + k2 * y↵
        ↪ [3], x[4] * k1 + k2 * y[4] }
215     z = normalize(z)
    self.position = z
    local p = { }
    p[0] = z[1]
    p[1] = z[2]
220     p[2] = z[3]
    p[3] = z[4]
    local o = { }
    o[0] = 0
    o[1] = 0
225     o[2] = 0
    o[3] = -1
    self.matrix1 = mquat2(o, qinv(p))
    local l = mmul(self.matrix1, { 0, 0, 0, 1})
    local q = { }
230     q[0] = l[1]
    q[1] = l[2]
    q[2] = l[3]
    q[3] = l[4]
    self.matrix2 = mquat2(qinv(q), o)
235     self.matrix3 = mquat2(qinv(p), o)
end

function sphere:in_1_list(atoms)
    local x = atoms[1]
240     local y = atoms[2]
    local z = atoms[3]
    local w = atoms[4]
    local m = self.angles
    if type(x) == "number" and type(y) == "number" and type(z) == "number" and ↵
        ↪ type(w) == "number" then
245         local n = self.quality
        local a = note2pol(x, y, z)
        local b = note2pol(y, z, w)
        local p = pol2car(a)
        local q = pol2car(b)
250         local d = delta(p, q, n)
        local s = mmul(self.matrix1, p)
        --      s      = mmul(self.matrix2, s)
        --      s      = mmul(self.matrix3, s)
        local t

```

```
255     local ss = project(s)
      for i = 1,n do
        t = mmul(self.matrix1, { p[1] + d[1] * i, p[2] + d[2] * i, p[3] + d[3] * i,
          ↵ , p[4] + d[4] * i })
      --   t = mmul(self.matrix2, t)
      --   t = mmul(self.matrix3, t)
260     local tt = project(t)
      if ss ~= nil and tt ~= nil then
        self:outlet(2, "list", ss)
        self:outlet(1, "list", tt)
      end
265     s = t
      ss = tt
    end
  end
end
```