

d01234

Claude Heiland-Allen

2007–2010

Contents

1	azidbazz~-help.pd	2
2	azidbazz~.pd	4
3	balls.pd	6
4	clap~.pd	28
5	COPYING	30
6	fmosc~.pd	36
7	Grid.lhs	37
8	hihat~.pd	49
9	hsv2rgb.pd	50
10	Hyperspace.hs	53
11	inkey.pd	55
12	kick~.pd	55
13	Makefile	56
14	mixtab-help.pd	56
15	mixtab.pd	56
16	nreceive-abs.pd	60
17	nreceive-help.pd	60
18	nreceive.pd	60
19	Polytopiary.lhs	61
20	psykaw~.pd	63
21	README	65
22	rotor.pd	65
23	seq256-gui-help.pd	66
24	seq256-gui.pd	67
25	snare~.pd	72
26	start.sh	76
27	textures/0.jpg	76
28	textures/0.png	77
29	textures/1.jpg	77
30	textures/1.png	78
31	textures/2.jpg	78
32	textures/2.png	79
33	textures/3.jpg	79
34	textures/3.png	80

1 azidbazz~-help.pd

```
#N canvas 20 34 343 401 10;
#X obj 116 302 dac~;
#X obj 126 272 azidbazz~;
#X msg 31 185 note $1;
```

```

5  #X floatatom 18 166 5 0 0 0 - - -;
   #X obj 26 64 f 0;
   #X obj 61 60 + 1;
   #X obj 24 17 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 1 1
   ;
10 #X msg 150 33 va \$1;
   #X floatatom 151 14 5 0 0 0 - - -;
   #X floatatom 151 54 5 0 0 0 - - -;
   #X floatatom 151 97 5 0 0 0 - - -;
   #X floatatom 151 155 5 0 0 0 - - -;
15 #X obj 149 113 / 100;
   #X msg 150 135 vs \$1;
   #X msg 150 174 vr \$1;
   #X msg 150 73 vd \$1;
   #X obj 183 244 t a;
20 #X floatatom 211 14 5 0 0 0 - - -;
   #X floatatom 211 54 5 0 0 0 - - -;
   #X floatatom 211 97 5 0 0 0 - - -;
   #X floatatom 211 155 5 0 0 0 - - -;
   #X floatatom 211 195 5 0 0 0 - - -;
25 #X msg 210 33 fa \$1;
   #X msg 210 73 fd \$1;
   #X msg 210 135 fs \$1;
   #X msg 210 174 fr \$1;
   #X msg 210 214 fp \$1;
30 #X msg 4 105 0;
   #X obj 27 101 select 0 1 2 3 4;
   #X obj 62 80 mod 6;
   #X msg 111 133 43;
   #X obj 209 113 / 10;
35 #X obj 25 40 metro 133;
   #X obj 5 82 delay 100;
   #X msg 18 131 36;
   #X msg 55 133 48;
   #X msg 81 133 43;
40 #X connect 1 0 0 0;
   #X connect 1 0 0 1;
   #X connect 2 0 1 0;
   #X connect 3 0 2 0;
   #X connect 4 0 5 0;
45 #X connect 4 0 28 0;
   #X connect 5 0 29 0;
   #X connect 6 0 32 0;
   #X connect 7 0 1 0;
   #X connect 8 0 7 0;
50 #X connect 9 0 15 0;
   #X connect 10 0 12 0;
   #X connect 11 0 14 0;
   #X connect 12 0 13 0;
   #X connect 13 0 1 0;
55 #X connect 14 0 1 0;
   #X connect 15 0 1 0;
   #X connect 16 0 1 0;
   #X connect 17 0 22 0;
   #X connect 18 0 23 0;
60 #X connect 19 0 31 0;
   #X connect 20 0 25 0;

```

```

#X connect 21 0 26 0;
#X connect 22 0 16 0;
#X connect 23 0 16 0;
65 #X connect 24 0 16 0;
#X connect 25 0 16 0;
#X connect 26 0 16 0;
#X connect 27 0 3 0;
#X connect 28 0 34 0;
70 #X connect 28 1 34 0;
#X connect 28 2 35 0;
#X connect 28 3 36 0;
#X connect 28 4 34 0;
#X connect 28 5 30 0;
75 #X connect 29 0 4 1;
#X connect 30 0 3 0;
#X connect 31 0 24 0;
#X connect 32 0 4 0;
#X connect 32 0 33 0;
80 #X connect 33 0 27 0;
#X connect 34 0 3 0;
#X connect 35 0 3 0;
#X connect 36 0 3 0;

```

2 azidbazz~.pd

```

#N canvas 33 58 469 563 10;
#X obj 12 17 inlet;
#X obj 12 101 select 0;
#X obj 12 146 0;
5 #X obj 13 178 vline~;
#X obj 63 123 t b f;
#X obj 37 146 1;
#X obj 93 172 vline~;
#X obj 93 147 pack f 10;
10 #X obj 92 198 mtof~;
#X obj 206 298 analogue_adsr~;
#X obj 109 278 *~;
#X obj 108 231 *~ 3;
#X obj 142 233 *~ 5;
15 #X obj 69 253 osc~;
#X obj 108 250 osc~;
#X obj 141 254 osc~;
#X obj 34 253 osc~;
#X obj 33 228 *~ 0.5;
20 #X obj 51 278 *~;
#X obj 72 301 +~;
#X msg 299 276 0.99;
#X msg 231 276 1;
#X obj 132 340 *~;
25 #X obj 393 219 loadbang;
#X obj 61 375 expr~ tanh($v1);
#X obj 61 402 lop~ 200;
#X obj 181 362 *~ 10;
#X obj 181 385 expr~ tanh($v1);
30 #X obj 11 41 route note na va vd vs vr fa fp fd fs fr q;
#X obj 181 407 vcf~ 8;
#X obj 131 455 expr~ tanh($v1);

```

```
#X obj 130 498 outlet~;
#X obj 316 298 analogue_adsr~;
35 #X msg 409 276 0.99;
#X msg 347 276 12;
#X obj 315 320 +~ 36;
#X obj 315 343 mtof~;
#X obj 191 231 sig~;
40 #X obj 200 205 sig~;
#X obj 220 185 sig~;
#X obj 230 165 sig~;
#X obj 311 231 sig~;
#X obj 335 190 sig~;
45 #X obj 355 170 sig~;
#X obj 365 150 sig~;
#X obj 321 211 sig~;
#X obj 62 340 *~;
#X connect 0 0 28 0;
50 #X connect 1 0 2 0;
#X connect 1 1 4 0;
#X connect 2 0 3 0;
#X connect 3 0 9 0;
#X connect 3 0 32 0;
55 #X connect 4 0 5 0;
#X connect 4 1 7 0;
#X connect 5 0 3 0;
#X connect 6 0 8 0;
#X connect 7 0 6 0;
60 #X connect 8 0 13 0;
#X connect 8 0 17 0;
#X connect 8 0 11 0;
#X connect 8 0 12 0;
#X connect 9 0 22 1;
65 #X connect 9 0 46 1;
#X connect 10 0 19 1;
#X connect 11 0 14 0;
#X connect 12 0 15 0;
#X connect 13 0 18 1;
70 #X connect 14 0 10 0;
#X connect 15 0 10 1;
#X connect 16 0 18 0;
#X connect 16 0 46 0;
#X connect 17 0 16 0;
75 #X connect 18 0 19 0;
#X connect 19 0 22 0;
#X connect 20 0 9 6;
#X connect 21 0 9 2;
#X connect 22 0 24 0;
80 #X connect 22 0 26 0;
#X connect 23 0 20 0;
#X connect 23 0 21 0;
#X connect 23 0 33 0;
#X connect 23 0 34 0;
85 #X connect 24 0 25 0;
#X connect 25 0 30 0;
#X connect 26 0 27 0;
#X connect 27 0 29 0;
#X connect 28 0 1 0;
```

```

90  #X connect 28 1 7 1;
    #X connect 28 2 37 0;
    #X connect 28 3 38 0;
    #X connect 28 4 39 0;
    #X connect 28 5 40 0;
95  #X connect 28 6 41 0;
    #X connect 28 7 45 0;
    #X connect 28 8 42 0;
    #X connect 28 9 43 0;
    #X connect 28 10 44 0;
100 #X connect 28 11 29 2;
    #X connect 29 0 30 0;
    #X connect 30 0 31 0;
    #X connect 32 0 35 0;
    #X connect 33 0 32 6;
105 #X connect 34 0 32 2;
    #X connect 35 0 36 0;
    #X connect 36 0 29 1;
    #X connect 37 0 9 1;
    #X connect 38 0 9 3;
110 #X connect 39 0 9 4;
    #X connect 40 0 9 5;
    #X connect 41 0 32 1;
    #X connect 42 0 32 3;
    #X connect 43 0 32 4;
115 #X connect 44 0 32 5;
    #X connect 45 0 32 2;
    #X connect 46 0 24 0;

```

3 balls.pd

```

#N canvas 0 0 1255 691 10;
#X obj 211 282 separator;
#X msg 34 54 destroy;
#X obj 51 177 loadbang;
5  #X obj 211 23 gemhead;
    #X obj 171 447 color;
    #X obj 211 364 translate;
    #X obj 240 312 loadbang;
    #X obj 401 114 pack f f f f f f;
10 #X msg 240 338 3;
    #X floatatom 442 138 5 0 0 0 - - -;
    #X obj 442 157 / 100;
    #X floatatom 482 138 5 0 0 0 - - -;
    #X obj 482 157 / 100;
15 #X obj 482 179 t b f;
    #X floatatom 522 138 5 0 0 0 - - -;
    #X obj 522 157 / 100;
    #X obj 522 179 t b f;
    #X floatatom 562 138 5 0 0 0 - - -;
20 #X obj 562 157 / 100;
    #X obj 562 179 t b f;
    #X obj 641 101 loadbang;
    #X msg 474 468 draw line;
    #X obj 211 145 repeat 256;
25 #X obj 170 557 cube;
    #X obj 348 167 f;

```

```
#X obj 380 178 + 1;
#N canvas 0 0 450 300 nop 0;
#X obj 29 16 inlet;
30 #X obj 29 273 outlet;
#X connect 0 0 1 0;
#X restore 321 387 pd nop;
#X msg 364 128 0;
#X msg 308 177 frame;
35 #X obj 211 174 t a b b;
#X msg 337 343 point \$1;
#X obj 347 200 t f f;
#X obj 507 277 pack f 1 1;
#X msg 240 248 size 0.1;
40 #X obj 239 225 loadbang;
#X obj 507 256 / 256;
#X obj 320 505 list prepend;
#N canvas 0 0 1131 439 \$0-audio 0;
#X obj 383 -5 inlet;
45 #X obj 175 299 outlet~;
#X obj 235 299 outlet~;
#X obj 415 -89 inlet;
#X obj 614 88 + 0;
#X obj 612 116 tabread \$0-seq;
50 #X obj 613 143 kick~;
#X obj 416 -65 t f f f f f f f f;
#X obj 722 116 tabread \$0-seq;
#X obj 723 143 snare~;
#X obj 724 88 + 32;
55 #X obj 832 116 tabread \$0-seq;
#X obj 833 143 hihat~;
#X obj 834 88 + 64;
#X obj 942 116 tabread \$0-seq;
#X obj 944 88 + 96;
60 #X obj 943 143 clap~;
#X obj 244 -35 + 224;
#X obj 244 -15 tabread \$0-seq;
#X obj 144 -15 tabread \$0-seq;
#X obj 144 -35 + 192;
65 #X obj 31 64 fmosc~ 128;
#X obj 108 65 fmosc~ 129;
#X obj 183 66 fmosc~ 130;
#X obj 259 68 fmosc~ 131;
#X obj 31 94 fmosc~ 192;
70 #X obj 108 95 fmosc~ 193;
#X obj 183 96 fmosc~ 194;
#X obj 259 98 fmosc~ 195;
#X obj 611 171 *~ 6;
#X obj 761 171 *~ 5;
75 #X obj 721 171 *~ 5;
#X obj 831 171 *~ 3;
#X obj 871 171 *~ 3;
#X obj 455 48 tabread \$0-seq;
#X obj 457 20 + 128;
80 #X obj 356 104 psyzaw~ 16;
#X obj 356 130 psyzaw~ 32;
#X obj 501 149 azidbazz~;
#X obj 499 22 tabread \$0-seq;
```

```
#X obj 499 0 + 160;
85 #X msg 501 129 note \$1;
#X obj 658 -84 inlet;
#X obj 501 71 t b f;
#X obj 501 93 delay 100;
#X msg 501 111 0;
90 #X obj 502 167 *~ 16;
#X obj 52 208 *~;
#X obj 102 208 *~;
#X obj 80 182 vline~;
#X obj 81 122 select 0;
95 #X msg 80 156 0 25;
#X msg 118 156 2 5;
#X obj 182 208 *~;
#X obj 232 208 *~;
#X obj 210 182 vline~;
100 #X obj 211 122 select 0;
#X msg 210 156 0 25;
#X msg 248 156 2 5;
#X obj 435 168 *~ 5;
#X obj 335 168 *~ 5;
105 #X obj 941 171 *~ 4;
#X obj 981 171 *~ 4;
#X connect 0 0 21 0;
#X connect 0 0 22 0;
#X connect 0 0 23 0;
110 #X connect 0 0 24 0;
#X connect 0 0 25 0;
#X connect 0 0 26 0;
#X connect 0 0 27 0;
#X connect 0 0 28 0;
115 #X connect 0 0 36 0;
#X connect 0 0 37 0;
#X connect 3 0 7 0;
#X connect 4 0 5 0;
#X connect 5 0 6 0;
120 #X connect 6 0 29 0;
#X connect 7 0 4 0;
#X connect 7 1 10 0;
#X connect 7 2 13 0;
#X connect 7 3 15 0;
125 #X connect 7 4 35 0;
#X connect 7 5 40 0;
#X connect 7 6 20 0;
#X connect 7 7 17 0;
#X connect 8 0 9 0;
130 #X connect 9 0 31 0;
#X connect 9 1 30 0;
#X connect 10 0 8 0;
#X connect 11 0 12 0;
#X connect 12 0 32 0;
135 #X connect 12 1 33 0;
#X connect 13 0 11 0;
#X connect 14 0 16 0;
#X connect 15 0 14 0;
#X connect 16 0 61 0;
140 #X connect 16 1 62 0;
```

```
#X connect 17 0 18 0;
#X connect 18 0 23 1;
#X connect 18 0 24 1;
#X connect 18 0 27 1;
145 #X connect 18 0 28 1;
#X connect 18 0 56 0;
#X connect 19 0 21 1;
#X connect 19 0 22 1;
#X connect 19 0 25 1;
150 #X connect 19 0 26 1;
#X connect 19 0 50 0;
#X connect 20 0 19 0;
#X connect 21 0 47 0;
#X connect 21 1 48 1;
155 #X connect 22 0 47 0;
#X connect 22 1 48 1;
#X connect 23 0 53 0;
#X connect 23 1 54 1;
#X connect 24 0 53 0;
160 #X connect 24 1 54 1;
#X connect 25 0 47 0;
#X connect 25 1 48 1;
#X connect 26 0 47 0;
#X connect 26 1 48 1;
165 #X connect 27 0 53 0;
#X connect 27 1 54 1;
#X connect 28 0 53 0;
#X connect 28 1 54 1;
#X connect 29 0 2 0;
170 #X connect 29 0 1 0;
#X connect 30 0 2 0;
#X connect 31 0 1 0;
#X connect 32 0 1 0;
#X connect 33 0 2 0;
175 #X connect 34 0 36 1;
#X connect 34 0 37 1;
#X connect 35 0 34 0;
#X connect 36 0 60 0;
#X connect 36 1 59 0;
180 #X connect 37 0 60 0;
#X connect 37 1 59 0;
#X connect 38 0 46 0;
#X connect 39 0 43 0;
#X connect 40 0 39 0;
185 #X connect 41 0 38 0;
#X connect 42 0 38 0;
#X connect 43 0 44 0;
#X connect 43 1 41 0;
#X connect 44 0 45 0;
190 #X connect 45 0 41 0;
#X connect 46 0 2 0;
#X connect 46 0 1 0;
#X connect 47 0 1 0;
#X connect 48 0 2 0;
195 #X connect 49 0 47 1;
#X connect 49 0 48 0;
#X connect 50 0 51 0;
```

```

#X connect 50 1 52 0;
#X connect 51 0 49 0;
200 #X connect 52 0 49 0;
#X connect 53 0 1 0;
#X connect 54 0 2 0;
#X connect 55 0 53 1;
#X connect 55 0 54 0;
205 #X connect 56 0 57 0;
#X connect 56 1 58 0;
#X connect 57 0 55 0;
#X connect 58 0 55 0;
#X connect 59 0 2 0;
210 #X connect 60 0 1 0;
#X connect 61 0 1 0;
#X connect 62 0 2 0;
#X restore 320 540 pd \$0-audio;
#X obj 320 563 *~ 0;
215 #X obj 390 563 *~ 0;
#X obj 347 589 dac~;
#X obj 452 581 / 100;
#X floatatom 452 561 5 0 0 0 - - -;
#X msg 18 21 create :0.1;
220 #X msg 327 281 fill \$1;
#X obj 636 283 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 0
1;
#X obj 636 370 f 0;
#X obj 670 372 + 1;
225 #X obj 636 319 metro 111;
#X msg 403 389 size \$1;
#X obj 669 394 mod 32;
#X obj 327 229 mod 32;
#X obj 18 242 gemwin 25;
230 #X obj 967 378 seq256-gui;
#X obj 967 633 s \$0-seq;
#X obj 403 343 * 0.01;
#X obj 403 323 tabread \$0-seq;
#X obj 1029 633 table \$0-seq 256;
235 #X obj 403 367 + 0.05;
#X obj 323 616 writesf~ 2;
#X msg 81 591 open -bytes 4 d01234.wav \, start;
#X msg 80 615 stop;
#X obj 44 558 select 0 1;
240 #X obj 44 531 tgl 15 0 empty empty empty 17 7 0 10 -262144 -1 -1 0
1;
#X obj 329 256 == -1;
#X msg 641 121 0;
#X obj 442 222 pack f f f f;
245 #X obj 395 245 mod 4;
#X msg 396 265 texture \$1;
#X obj 321 408 route translate color size fill texture;
#X obj 170 501 pix_texture;
#X obj 170 478 pix_multiimage;
250 #X msg 38 360 open textures/*.jpg 3;
#X msg 35 395 mode 0;
#X obj 9 332 loadbang;
#X msg 375 470 draw quads;
#X msg 51 201 reset \, dimen 640 480 \, border 0;

```

```
255 #X msg 507 369 color \$1 \$2 \$3;
    #X obj 456 434 select 1;
    #X obj 211 71 t a b b;
    #X obj 452 15 t b b b b b b;
    #X obj 644 47 rotor;
260 #X msg 99 55 create;
    #X obj 321 304 hsect Hyperspace 4;
    #X obj 507 300 hsv2rgb;
    #X msg 108 143 border 1;
    #X obj 999 3 mixtab 15;
265 #N canvas 25 81 593 308 \$0-control 0;
    #X obj 129 6 inlet;
    #X obj 264 276 outlet;
    #X obj 211 8 inlet;
    #X obj 298 5 inlet;
270 #N canvas 0 0 1255 691 \$0-channel-A 0;
    #X obj 558 12 inlet;
    #X obj 300 659 outlet;
    #X obj 429 343 unpack f f;
    #X obj 444 366 swap;
275 #X obj 428 423 route 16 17 18 19 20 21 22 23;
    #X obj 430 400 pack f f;
    #X obj 11 492 spigot;
    #X obj 481 376 == 0;
    #X obj 91 492 spigot;
280 #X obj 171 492 spigot;
    #X obj 251 492 spigot;
    #X obj 331 492 spigot;
    #X obj 411 492 spigot;
    #X obj 491 492 spigot;
285 #X obj 571 492 spigot;
    #X obj 12 29 unpack f f;
    #X obj 29 60 swap;
    #X obj 15 85 pack f f;
    #X obj 11 113 route 24 25 26 27;
290 #X msg 11 140 96 \$1;
    #X msg 52 140 97 \$1;
    #X msg 91 140 98 \$1;
    #X msg 131 140 99 \$1;
    #X obj 11 426 t a a a a a a a;
295 #X obj 571 513 unpack f f;
    #X obj 571 558 pack f f;
    #X obj 491 513 unpack f f;
    #X obj 491 558 pack f f;
    #X obj 411 513 unpack f f;
300 #X obj 411 558 pack f f;
    #X obj 331 513 unpack f f;
    #X obj 331 558 pack f f;
    #X obj 251 513 unpack f f;
    #X obj 251 558 pack f f;
305 #X obj 171 513 unpack f f;
    #X obj 171 558 pack f f;
    #X obj 91 513 unpack f f;
    #X obj 91 558 pack f f;
    #X obj 11 513 unpack f f;
310 #X obj 11 558 pack f f;
    #X obj 11 535 + 0;
```

```

#X obj 91 535 + 4;
#X obj 171 535 + 8;
#X obj 251 535 + 12;
315 #X obj 331 535 + 16;
#X obj 411 535 + 20;
#X obj 491 535 + 24;
#X obj 571 535 + 28;
#X obj 300 621 t a;
320 #X obj 177 113 route 32 33 34 35;
#X msg 177 140 64 \$1;
#X msg 218 140 65 \$1;
#X msg 257 140 66 \$1;
#X msg 297 140 67 \$1;
325 #X obj 176 174 t a;
#X obj 345 173 t a;
#X obj 346 112 route 40 41 42 43;
#X msg 346 139 32 \$1;
#X msg 387 139 33 \$1;
330 #X msg 426 139 34 \$1;
#X msg 466 139 35 \$1;
#X obj 516 169 t a;
#X obj 516 112 route 66 67 68 69;
#X msg 516 139 0 \$1;
335 #X msg 557 139 1 \$1;
#X msg 596 139 2 \$1;
#X msg 636 138 3 \$1;
#X obj 11 174 t a;
#X obj 853 173 t a;
340 #X obj 1021 172 t a;
#X obj 1192 168 t a;
#X obj 687 173 t a;
#X obj 687 112 route 28 29 30 31;
#X obj 853 112 route 36 37 38 39;
345 #X obj 1022 111 route 44 45 46 47;
#X obj 1192 111 route 70 71 72 73;
#X msg 687 139 96 \$1;
#X msg 728 139 97 \$1;
#X msg 767 139 98 \$1;
350 #X msg 807 139 99 \$1;
#X msg 853 139 64 \$1;
#X msg 894 139 65 \$1;
#X msg 933 139 66 \$1;
#X msg 973 139 67 \$1;
355 #X msg 1022 138 32 \$1;
#X msg 1063 138 33 \$1;
#X msg 1102 138 34 \$1;
#X msg 1142 138 35 \$1;
#X msg 1192 138 0 \$1;
360 #X msg 1233 138 1 \$1;
#X msg 1272 138 2 \$1;
#X msg 1311 138 3 \$1;
#X obj 1051 346 unpack f f;
#X obj 1066 369 swap;
365 #X obj 1050 426 route 16 17 18 19 20 21 22 23;
#X obj 1052 403 pack f f;
#X obj 661 492 spigot;
#X obj 741 492 spigot;

```

```

#X obj 821 492 spigot;
370 #X obj 901 492 spigot;
#X obj 981 492 spigot;
#X obj 1061 492 spigot;
#X obj 1141 492 spigot;
#X obj 1221 492 spigot;
375 #X obj 661 426 t a a a a a a a a;
#X obj 1221 513 unpack f f;
#X obj 1221 558 pack f f;
#X obj 1141 513 unpack f f;
#X obj 1141 558 pack f f;
380 #X obj 1061 513 unpack f f;
#X obj 1061 558 pack f f;
#X obj 981 513 unpack f f;
#X obj 981 558 pack f f;
#X obj 901 513 unpack f f;
385 #X obj 901 558 pack f f;
#X obj 821 513 unpack f f;
#X obj 821 558 pack f f;
#X obj 741 513 unpack f f;
#X obj 741 558 pack f f;
390 #X obj 661 513 unpack f f;
#X obj 661 558 pack f f;
#X obj 661 535 + 0;
#X obj 741 535 + 4;
#X obj 901 535 + 12;
395 #X obj 981 535 + 16;
#X obj 1061 535 + 20;
#X obj 1141 535 + 24;
#X obj 1221 535 + 28;
#X obj 950 621 t a;
400 #X obj 1103 377 != 0;
#X obj 823 534 + 8;
#X connect 0 0 2 0;
#X connect 0 0 15 0;
#X connect 0 0 92 0;
405 #X connect 2 0 3 0;
#X connect 2 1 3 1;
#X connect 3 0 5 0;
#X connect 3 1 7 0;
#X connect 4 0 6 1;
410 #X connect 4 1 8 1;
#X connect 4 2 9 1;
#X connect 4 3 10 1;
#X connect 4 4 11 1;
#X connect 4 5 12 1;
415 #X connect 4 6 13 1;
#X connect 4 7 14 1;
#X connect 5 0 4 0;
#X connect 6 0 38 0;
#X connect 7 0 5 1;
420 #X connect 8 0 36 0;
#X connect 9 0 34 0;
#X connect 10 0 32 0;
#X connect 11 0 30 0;
#X connect 12 0 28 0;
425 #X connect 13 0 26 0;

```

```
#X connect 14 0 24 0;
#X connect 15 0 16 0;
#X connect 15 1 16 1;
#X connect 16 0 17 0;
430 #X connect 16 1 17 1;
#X connect 17 0 18 0;
#X connect 18 0 19 0;
#X connect 18 1 20 0;
#X connect 18 2 21 0;
435 #X connect 18 3 22 0;
#X connect 18 4 49 0;
#X connect 19 0 67 0;
#X connect 20 0 67 0;
#X connect 21 0 67 0;
440 #X connect 22 0 67 0;
#X connect 23 0 6 0;
#X connect 23 1 8 0;
#X connect 23 2 9 0;
#X connect 23 3 10 0;
445 #X connect 23 4 11 0;
#X connect 23 5 12 0;
#X connect 23 6 13 0;
#X connect 23 7 14 0;
#X connect 24 0 47 0;
450 #X connect 24 1 25 1;
#X connect 25 0 48 0;
#X connect 26 0 46 0;
#X connect 26 1 27 1;
#X connect 27 0 48 0;
455 #X connect 28 0 45 0;
#X connect 28 1 29 1;
#X connect 29 0 48 0;
#X connect 30 0 44 0;
#X connect 30 1 31 1;
460 #X connect 31 0 48 0;
#X connect 32 0 43 0;
#X connect 32 1 33 1;
#X connect 33 0 48 0;
#X connect 34 0 42 0;
465 #X connect 34 1 35 1;
#X connect 35 0 48 0;
#X connect 36 0 41 0;
#X connect 36 1 37 1;
#X connect 37 0 48 0;
470 #X connect 38 0 40 0;
#X connect 38 1 39 1;
#X connect 39 0 48 0;
#X connect 40 0 39 0;
#X connect 41 0 37 0;
475 #X connect 42 0 35 0;
#X connect 43 0 33 0;
#X connect 44 0 31 0;
#X connect 45 0 29 0;
#X connect 46 0 27 0;
480 #X connect 47 0 25 0;
#X connect 48 0 1 0;
#X connect 49 0 50 0;
```

```
#X connect 49 1 51 0;
#X connect 49 2 52 0;
485 #X connect 49 3 53 0;
#X connect 49 4 56 0;
#X connect 50 0 54 0;
#X connect 51 0 54 0;
#X connect 52 0 54 0;
490 #X connect 53 0 54 0;
#X connect 54 0 23 0;
#X connect 55 0 23 0;
#X connect 56 0 57 0;
#X connect 56 1 58 0;
495 #X connect 56 2 59 0;
#X connect 56 3 60 0;
#X connect 56 4 62 0;
#X connect 57 0 55 0;
#X connect 58 0 55 0;
500 #X connect 59 0 55 0;
#X connect 60 0 55 0;
#X connect 61 0 23 0;
#X connect 62 0 63 0;
#X connect 62 1 64 0;
505 #X connect 62 2 65 0;
#X connect 62 3 66 0;
#X connect 62 4 72 0;
#X connect 63 0 61 0;
#X connect 64 0 61 0;
510 #X connect 65 0 61 0;
#X connect 66 0 61 0;
#X connect 67 0 23 0;
#X connect 68 0 104 0;
#X connect 69 0 104 0;
515 #X connect 70 0 104 0;
#X connect 71 0 104 0;
#X connect 72 0 76 0;
#X connect 72 1 77 0;
#X connect 72 2 78 0;
520 #X connect 72 3 79 0;
#X connect 72 4 73 0;
#X connect 73 0 80 0;
#X connect 73 1 81 0;
#X connect 73 2 82 0;
525 #X connect 73 3 83 0;
#X connect 73 4 74 0;
#X connect 74 0 84 0;
#X connect 74 1 85 0;
#X connect 74 2 86 0;
530 #X connect 74 3 87 0;
#X connect 74 4 75 0;
#X connect 75 0 88 0;
#X connect 75 1 89 0;
#X connect 75 2 90 0;
535 #X connect 75 3 91 0;
#X connect 76 0 71 0;
#X connect 77 0 71 0;
#X connect 78 0 71 0;
#X connect 79 0 71 0;
```

```
540  #X connect 80 0 68 0;
      #X connect 81 0 68 0;
      #X connect 82 0 68 0;
      #X connect 83 0 68 0;
      #X connect 84 0 69 0;
545  #X connect 85 0 69 0;
      #X connect 86 0 69 0;
      #X connect 87 0 69 0;
      #X connect 88 0 70 0;
      #X connect 89 0 70 0;
550  #X connect 90 0 70 0;
      #X connect 91 0 70 0;
      #X connect 92 0 93 0;
      #X connect 92 1 93 1;
      #X connect 93 0 95 0;
555  #X connect 93 1 129 0;
      #X connect 94 0 96 1;
      #X connect 94 1 97 1;
      #X connect 94 2 98 1;
      #X connect 94 3 99 1;
560  #X connect 94 4 100 1;
      #X connect 94 5 101 1;
      #X connect 94 6 102 1;
      #X connect 94 7 103 1;
      #X connect 95 0 94 0;
565  #X connect 96 0 119 0;
      #X connect 97 0 117 0;
      #X connect 98 0 115 0;
      #X connect 99 0 113 0;
      #X connect 100 0 111 0;
570  #X connect 101 0 109 0;
      #X connect 102 0 107 0;
      #X connect 103 0 105 0;
      #X connect 104 0 96 0;
      #X connect 104 1 97 0;
575  #X connect 104 2 98 0;
      #X connect 104 3 99 0;
      #X connect 104 4 100 0;
      #X connect 104 5 101 0;
      #X connect 104 6 102 0;
580  #X connect 104 7 103 0;
      #X connect 105 0 127 0;
      #X connect 105 1 106 1;
      #X connect 106 0 128 0;
      #X connect 107 0 126 0;
585  #X connect 107 1 108 1;
      #X connect 108 0 128 0;
      #X connect 109 0 125 0;
      #X connect 109 1 110 1;
      #X connect 110 0 128 0;
590  #X connect 111 0 124 0;
      #X connect 111 1 112 1;
      #X connect 112 0 128 0;
      #X connect 113 0 123 0;
      #X connect 113 1 114 1;
595  #X connect 114 0 128 0;
      #X connect 115 0 130 0;
```

```
#X connect 115 1 116 1;
#X connect 116 0 128 0;
#X connect 117 0 122 0;
600 #X connect 117 1 118 1;
#X connect 118 0 128 0;
#X connect 119 0 121 0;
#X connect 119 1 120 1;
#X connect 120 0 128 0;
605 #X connect 121 0 120 0;
#X connect 122 0 118 0;
#X connect 123 0 114 0;
#X connect 124 0 112 0;
#X connect 125 0 110 0;
610 #X connect 126 0 108 0;
#X connect 127 0 106 0;
#X connect 128 0 1 0;
#X connect 129 0 95 1;
#X connect 130 0 116 0;
615 #X restore 264 226 pd \${0-channel-A};
#X obj 65 275 outlet;
#N canvas 0 0 1255 691 \${0-channel-B 0};
#X obj 8 6 inlet;
#X obj 379 517 outlet;
620 #X obj 8 28 unpack f f;
#X obj 25 59 swap;
#X obj 11 84 pack f f;
#X obj 11 113 route 24 32 40 66;
#X msg 96 204 bot \${1};
625 #X msg 68 182 top \${1};
#X msg 39 161 mul \${1};
#X msg 11 140 add \${1};
#X obj 11 231 list prepend 0;
#X msg 226 204 bot \${1};
630 #X msg 198 182 top \${1};
#X msg 169 161 mul \${1};
#X msg 141 140 add \${1};
#X msg 356 204 bot \${1};
#X msg 328 182 top \${1};
635 #X msg 299 161 mul \${1};
#X msg 271 140 add \${1};
#X msg 486 204 bot \${1};
#X msg 458 182 top \${1};
#X msg 429 161 mul \${1};
640 #X msg 401 140 add \${1};
#X msg 616 204 bot \${1};
#X msg 588 182 top \${1};
#X msg 559 161 mul \${1};
#X msg 531 140 add \${1};
645 #X msg 746 204 bot \${1};
#X msg 718 182 top \${1};
#X msg 689 161 mul \${1};
#X msg 661 140 add \${1};
#X text 825 169 rotation;
650 #X obj 141 113 route 25 33 41 67;
#X obj 141 231 list prepend 1;
#X obj 379 472 t a;
#X obj 271 113 route 26 34 42 68;
```

```
#X obj 271 231 list prepend 2;
655 #X obj 401 113 route 27 35 43 69;
#X obj 401 231 list prepend 3;
#X obj 531 231 list prepend 4;
#X obj 661 231 list prepend 5;
#X obj 531 113 route 28 36 44 70;
660 #X obj 661 113 route 29 37 45 71;
#X obj 667 305 route 16 17 18 19;
#X msg 667 342 6 \$1;
#X msg 707 342 7 \$1;
#X msg 747 342 8 \$1;
665 #X msg 787 342 9 \$1;
#X text 835 315 scales;
#X connect 0 0 2 0;
#X connect 2 0 3 0;
#X connect 2 1 3 1;
670 #X connect 3 0 4 0;
#X connect 3 1 4 1;
#X connect 4 0 5 0;
#X connect 5 0 9 0;
#X connect 5 1 8 0;
675 #X connect 5 2 7 0;
#X connect 5 3 6 0;
#X connect 5 4 32 0;
#X connect 6 0 10 0;
#X connect 7 0 10 0;
680 #X connect 8 0 10 0;
#X connect 9 0 10 0;
#X connect 10 0 34 0;
#X connect 11 0 33 0;
#X connect 12 0 33 0;
685 #X connect 13 0 33 0;
#X connect 14 0 33 0;
#X connect 15 0 36 0;
#X connect 16 0 36 0;
#X connect 17 0 36 0;
690 #X connect 18 0 36 0;
#X connect 19 0 38 0;
#X connect 20 0 38 0;
#X connect 21 0 38 0;
#X connect 22 0 38 0;
695 #X connect 23 0 39 0;
#X connect 24 0 39 0;
#X connect 25 0 39 0;
#X connect 26 0 39 0;
#X connect 27 0 40 0;
700 #X connect 28 0 40 0;
#X connect 29 0 40 0;
#X connect 30 0 40 0;
#X connect 32 0 14 0;
#X connect 32 1 13 0;
705 #X connect 32 2 12 0;
#X connect 32 3 11 0;
#X connect 32 4 35 0;
#X connect 33 0 34 0;
#X connect 34 0 1 0;
710 #X connect 35 0 18 0;
```

```
#X connect 35 1 17 0;
#X connect 35 2 16 0;
#X connect 35 3 15 0;
#X connect 35 4 37 0;
715 #X connect 36 0 34 0;
#X connect 37 0 22 0;
#X connect 37 1 21 0;
#X connect 37 2 20 0;
#X connect 37 3 19 0;
720 #X connect 37 4 41 0;
#X connect 38 0 34 0;
#X connect 39 0 34 0;
#X connect 40 0 34 0;
#X connect 41 0 26 0;
725 #X connect 41 1 25 0;
#X connect 41 2 24 0;
#X connect 41 3 23 0;
#X connect 41 4 42 0;
#X connect 42 0 30 0;
730 #X connect 42 1 29 0;
#X connect 42 2 28 0;
#X connect 42 3 27 0;
#X connect 42 4 43 0;
#X connect 43 0 44 0;
735 #X connect 43 1 45 0;
#X connect 43 2 46 0;
#X connect 43 3 47 0;
#X connect 44 0 34 0;
#X connect 45 0 34 0;
740 #X connect 46 0 34 0;
#X connect 47 0 34 0;
#X restore 65 219 pd \${0-channel-B;
#X obj 18 5 inlet;
#N canvas 0 0 450 300 \${0-keyboard 0;
745 #X obj 15 21 inlet;
#X obj 21 259 outlet;
#X obj 15 41 route keypress;
#X obj 15 61 unpack f f f s;
#N canvas 17 29 850 231 \${0-key2midi 0;
750 #X obj 11 17 inlet;
#X obj 11 194 outlet;
#X obj 11 116 0;
#X obj 34 116 36;
#X obj 57 116 37;
755 #X obj 80 116 38;
#X obj 103 116 39;
#X obj 126 116 40;
#X obj 149 116 41;
#X obj 172 116 42;
760 #X obj 195 116 43;
#X obj 218 116 44;
#X obj 241 116 45;
#X obj 264 116 46;
#X obj 287 116 47;
765 #X obj 310 116 48;
#X obj 333 116 49;
#X obj 356 116 50;
```

```

#X obj 379 116 51;
#X obj 11 61 select space z s x d c v g b h n j m comma l period semicolon
770 slash;
#X obj 402 116 52;
#X obj 387 59 select q D2 w D3 e r D5 t D6 y D7 u i D9 o D0 p bracketleft
equal bracketright;
#X obj 425 116 53;
775 #X obj 448 116 54;
#X obj 471 116 55;
#X obj 494 116 56;
#X obj 517 116 57;
#X obj 540 116 58;
780 #X obj 563 116 59;
#X obj 586 116 60;
#X obj 609 116 61;
#X obj 632 116 62;
#X obj 655 116 63;
785 #X obj 678 116 64;
#X obj 701 116 65;
#X obj 724 116 66;
#X obj 747 116 67;
#X obj 795 190 outlet;
790 #X connect 0 0 19 0;
#X connect 2 0 1 0;
#X connect 3 0 1 0;
#X connect 4 0 1 0;
#X connect 5 0 1 0;
795 #X connect 6 0 1 0;
#X connect 7 0 1 0;
#X connect 8 0 1 0;
#X connect 9 0 1 0;
#X connect 10 0 1 0;
800 #X connect 11 0 1 0;
#X connect 12 0 1 0;
#X connect 13 0 1 0;
#X connect 14 0 1 0;
#X connect 15 0 1 0;
805 #X connect 16 0 1 0;
#X connect 17 0 1 0;
#X connect 18 0 1 0;
#X connect 19 0 2 0;
#X connect 19 1 3 0;
810 #X connect 19 2 4 0;
#X connect 19 3 5 0;
#X connect 19 4 6 0;
#X connect 19 5 7 0;
#X connect 19 6 8 0;
815 #X connect 19 7 9 0;
#X connect 19 8 10 0;
#X connect 19 9 11 0;
#X connect 19 10 12 0;
#X connect 19 11 13 0;
820 #X connect 19 12 14 0;
#X connect 19 13 15 0;
#X connect 19 14 16 0;
#X connect 19 15 17 0;
#X connect 19 16 18 0;

```

```
825 #X connect 19 17 20 0;
    #X connect 19 18 21 0;
    #X connect 20 0 1 0;
    #X connect 21 0 15 0;
    #X connect 21 1 16 0;
830 #X connect 21 2 17 0;
    #X connect 21 3 18 0;
    #X connect 21 4 20 0;
    #X connect 21 5 22 0;
    #X connect 21 6 23 0;
835 #X connect 21 7 24 0;
    #X connect 21 8 25 0;
    #X connect 21 9 26 0;
    #X connect 21 10 27 0;
    #X connect 21 11 28 0;
840 #X connect 21 12 29 0;
    #X connect 21 13 30 0;
    #X connect 21 14 31 0;
    #X connect 21 15 32 0;
    #X connect 21 16 33 0;
845 #X connect 21 17 34 0;
    #X connect 21 18 35 0;
    #X connect 21 19 36 0;
    #X connect 21 20 37 0;
    #X connect 22 0 1 0;
850 #X connect 23 0 1 0;
    #X connect 24 0 1 0;
    #X connect 25 0 1 0;
    #X connect 26 0 1 0;
    #X connect 27 0 1 0;
855 #X connect 28 0 1 0;
    #X connect 29 0 1 0;
    #X connect 30 0 1 0;
    #X connect 31 0 1 0;
    #X connect 32 0 1 0;
860 #X connect 33 0 1 0;
    #X connect 34 0 1 0;
    #X connect 35 0 1 0;
    #X connect 36 0 1 0;
    #X restore 108 117 pd \${0}-key2midi;
865 #N canvas 20 29 597 409 \${0}-cursor 0;
    #X obj 16 18 inlet;
    #X text 62 19 note;
    #X obj 186 18 inlet;
    #X text 238 19 key;
870 #X obj 186 45 select Return Right BackSpace Left Up Down F1 F2 F3 F4
    F5 F6 F7 F8;
    #X obj 43 174 + 1;
    #X obj 42 277 mod 32;
    #X obj 16 104 v \${0}-cursor-v;
875 #X obj 42 302 v \${0}-cursor-v;
    #X obj 16 55 t b f;
    #X obj 16 373 outlet;
    #X obj 80 375 outlet;
    #X obj 184 114 v \${0}-cursor-v;
880 #X obj 181 171 - 1;
    #X obj 179 144 v \${0}-cursor-v;
```

```
#X obj 217 175 v \${0-cursor-v};
#X obj 215 199 + 16;
#X obj 343 103 0;
885 #X obj 373 103 4;
#X obj 403 103 8;
#X obj 433 103 12;
#X obj 463 103 16;
#X obj 493 103 20;
890 #X obj 523 103 24;
#X obj 553 103 28;
#X obj 445 166 f;
#X obj 205 371 outlet;
#X obj 544 379 outlet;
895 #X obj 16 132 t f f;
#X connect 0 0 9 0;
#X connect 2 0 4 0;
#X connect 4 0 12 0;
#X connect 4 1 12 0;
900 #X connect 4 2 14 0;
#X connect 4 3 14 0;
#X connect 4 4 15 0;
#X connect 4 5 15 0;
#X connect 4 6 17 0;
905 #X connect 4 7 18 0;
#X connect 4 8 19 0;
#X connect 4 9 20 0;
#X connect 4 10 21 0;
#X connect 4 11 22 0;
910 #X connect 4 12 23 0;
#X connect 4 13 24 0;
#X connect 4 14 27 0;
#X connect 5 0 6 0;
#X connect 6 0 8 0;
915 #X connect 6 0 26 0;
#X connect 7 0 28 0;
#X connect 9 0 7 0;
#X connect 9 1 11 0;
#X connect 12 0 5 0;
920 #X connect 13 0 6 0;
#X connect 14 0 13 0;
#X connect 15 0 16 0;
#X connect 16 0 6 0;
#X connect 17 0 25 0;
925 #X connect 18 0 25 0;
#X connect 19 0 25 0;
#X connect 20 0 25 0;
#X connect 21 0 25 0;
#X connect 22 0 25 0;
930 #X connect 23 0 25 0;
#X connect 24 0 25 0;
#X connect 25 0 6 0;
#X connect 28 0 10 0;
#X connect 28 1 5 0;
935 #X restore 107 151 pd \${0-cursor};
#X obj 213 253 outlet;
#N canvas 43 337 450 300 \${0-instrument} 0;
#X obj 31 22 inlet;
```

```
#X text 30 3 raw cursor;
940 #X obj 141 24 inlet;
    #X obj 242 27 inlet;
    #X text 142 6 note;
    #X text 245 11 key;
    #X obj 31 178 + 128;
945 #X obj 31 221 pack f f;
    #X obj 242 56 select F9 F10 F11 F12;
    #X obj 245 82 128;
    #X obj 272 82 160;
    #X obj 305 81 192;
950 #X obj 348 83 224;
    #X obj 31 261 outlet;
    #X obj 384 259 outlet;
    #X connect 0 0 6 0;
    #X connect 2 0 7 1;
955 #X connect 3 0 8 0;
    #X connect 6 0 7 0;
    #X connect 7 0 13 0;
    #X connect 8 0 9 0;
    #X connect 8 1 10 0;
960 #X connect 8 2 11 0;
    #X connect 8 3 12 0;
    #X connect 8 4 14 0;
    #X connect 9 0 6 1;
    #X connect 10 0 6 1;
965 #X connect 11 0 6 1;
    #X connect 12 0 6 1;
    #X restore 108 183 pd \$0-instrument;
    #X obj 214 210 select Escape;
    #X obj 103 258 outlet;
970 #X connect 0 0 2 0;
    #X connect 2 0 3 0;
    #X connect 3 3 4 0;
    #X connect 4 0 5 0;
    #X connect 4 1 5 1;
975 #X connect 5 0 7 0;
    #X connect 5 1 7 1;
    #X connect 5 2 6 0;
    #X connect 5 3 7 2;
    #X connect 7 0 1 0;
980 #X connect 7 1 8 0;
    #X connect 8 0 9 0;
    #X restore 19 86 pd \$0-keyboard;
    #X obj 215 275 outlet;
    #X obj 7 269 outlet;
985 #X obj 351 66 unpack f f;
    #X obj 351 94 swap;
    #X obj 351 145 route 74 75 48;
    #X obj 350 120 pack f f;
    #X msg 292 183 filter \$1;
990 #X msg 372 183 decay \$1;
    #X obj 441 179 + 60;
    #X msg 444 206 speed \$1;
    #X obj 130 274 outlet;
    #X connect 0 0 4 0;
995 #X connect 0 0 11 0;
```

```

#X connect 2 0 6 0;
#X connect 2 0 11 0;
#X connect 3 0 11 0;
#X connect 4 0 1 0;
1000 #X connect 6 0 5 0;
#X connect 7 0 8 0;
#X connect 8 0 1 0;
#X connect 8 1 10 0;
#X connect 8 2 9 0;
1005 #X connect 11 0 12 0;
#X connect 11 1 12 1;
#X connect 12 0 14 0;
#X connect 12 1 14 1;
#X connect 13 0 15 0;
1010 #X connect 13 1 16 0;
#X connect 13 2 17 0;
#X connect 14 0 13 0;
#X connect 15 0 19 0;
#X connect 16 0 19 0;
1015 #X connect 17 0 18 0;
#X connect 18 0 19 0;
#X restore 881 351 pd \ $0-control;
#X obj 999 128 mixtab 16;
#X obj 999 253 mixtab 1;
1020 #X obj 560 14 route 0 1 2 3 4 5;
#X msg 560 -6 0 bpm \ $1 \, 1 bpm \ $1 \, 2 bpm \ $1 \, 3 bpm \ $1 \, 4
bpm \ $1 \, 5 bpm \ $1;
#X floatatom 671 281 5 0 0 0 - - -;
#X obj 670 231 loadbang;
1025 #X msg 670 255 130;
#X obj 394 47 rotor;
#X obj 444 47 rotor;
#X obj 494 47 rotor;
#X obj 544 47 rotor;
1030 #X obj 594 47 rotor;
#X obj 537 112 route 6 7 8 9;
#X msg 452 537 2.5;
#X obj 452 514 loadbang;
#X obj 752 50 #out window \, title d0-d1-d2-d3-d4;
1035 #X obj 920 382 nbx 2 20 -1e+37 1e+37 0 0 empty empty empty 0 -8 1 18
-241291 -1 -1 11 256;
#N canvas 0 0 450 300 \ $0-azidbazz-settings 0;
#X obj 25 266 outlet;
#X obj 30 47 inlet;
1040 #X obj 102 50 inlet;
#X msg 100 186 fp \ $1;
#X msg 26 115 va 6 \, vd 66 \, vs 0.6 \, vr 100 \, fa 6 \, fp 36 \,
fd 100 \, fs 12 \, fr 100;
#X obj 162 50 inlet;
1045 #X msg 160 186 fd \ $1;
#X obj 101 163 / 1.5;
#X connect 1 0 4 0;
#X connect 2 0 7 0;
#X connect 3 0 0 0;
1050 #X connect 4 0 0 0;
#X connect 5 0 6 0;
#X connect 6 0 0 0;

```

```
#X connect 7 0 3 0;
#X restore 434 609 pd \${0-azidbazz-settings};
1055 #X floatatom 511 577 5 0 0 0 - - -;
#X floatatom 551 577 5 0 0 0 - - -;
#X obj 730 390 route filter decay speed;
#X obj 822 79 ctlin;
#X obj 807 108 pack f f f;
1060 #X obj 752 136 list prepend INPUT:GRIDFLOW;
#X obj 807 160 list prepend INPUT:MIDI;
#X obj 752 194 t a b b;
#X obj 774 218 timer;
#X obj 752 249 list prepend;
1065 #X msg 55 91 1 \; pd dsp 1;
#X msg 38 126 0 \; pd dsp 0;
#X obj 123 91 select 1 0;
#X obj 751 274 print EVENT;
#X connect 0 0 5 0;
1070 #X connect 1 0 52 0;
#X connect 2 0 76 0;
#X connect 3 0 79 0;
#X connect 4 0 71 0;
#X connect 5 0 4 0;
1075 #X connect 6 0 8 0;
#X connect 7 0 83 2;
#X connect 8 0 5 1;
#X connect 9 0 10 0;
#X connect 10 0 66 0;
1080 #X connect 11 0 12 0;
#X connect 12 0 13 0;
#X connect 13 0 66 0;
#X connect 13 1 66 1;
#X connect 14 0 15 0;
1085 #X connect 15 0 16 0;
#X connect 16 0 66 0;
#X connect 16 1 66 2;
#X connect 17 0 18 0;
#X connect 18 0 19 0;
1090 #X connect 19 0 66 0;
#X connect 19 1 66 3;
#X connect 20 0 65 0;
#X connect 21 0 23 0;
#X connect 22 0 29 0;
1095 #X connect 24 0 25 0;
#X connect 24 0 31 0;
#X connect 25 0 24 1;
#X connect 26 0 69 0;
#X connect 27 0 24 1;
1100 #X connect 28 0 26 0;
#X connect 29 0 0 0;
#X connect 29 1 83 0;
#X connect 29 2 24 0;
#X connect 30 0 26 0;
1105 #X connect 31 0 35 0;
#X connect 31 0 51 0;
#X connect 31 0 56 0;
#X connect 31 0 67 0;
#X connect 31 1 30 0;
```

```
1110  #X connect 31 1 36 1;
      #X connect 32 0 84 0;
      #X connect 33 0 26 0;
      #X connect 34 0 33 0;
      #X connect 35 0 32 0;
1115  #X connect 36 0 37 0;
      #X connect 37 0 38 0;
      #X connect 37 1 39 0;
      #X connect 38 0 40 0;
      #X connect 38 0 59 0;
1120  #X connect 39 0 40 1;
      #X connect 39 0 59 1;
      #X connect 41 0 39 1;
      #X connect 41 0 38 1;
      #X connect 42 0 41 0;
1125  #X connect 43 0 52 0;
      #X connect 44 0 26 0;
      #X connect 45 0 48 0;
      #X connect 45 0 118 0;
      #X connect 46 0 47 0;
1130  #X connect 46 0 37 1;
      #X connect 46 0 64 1;
      #X connect 47 0 50 0;
      #X connect 48 0 46 0;
      #X connect 49 0 26 0;
1135  #X connect 50 0 46 1;
      #X connect 51 0 64 0;
      #X connect 53 0 54 0;
      #X connect 55 0 58 0;
      #X connect 56 0 55 0;
1140  #X connect 58 0 49 0;
      #X connect 60 0 59 0;
      #X connect 61 0 59 0;
      #X connect 62 0 61 0;
      #X connect 62 1 60 0;
1145  #X connect 63 0 62 0;
      #X connect 64 0 44 0;
      #X connect 65 0 17 0;
      #X connect 65 0 14 0;
      #X connect 65 0 11 0;
1150  #X connect 65 0 9 0;
      #X connect 66 0 83 3;
      #X connect 67 0 68 0;
      #X connect 68 0 26 0;
      #X connect 69 0 5 2;
1155  #X connect 69 0 36 0;
      #X connect 69 1 4 1;
      #X connect 69 2 23 1;
      #X connect 69 3 78 0;
      #X connect 69 4 71 1;
1160  #X connect 70 0 23 0;
      #X connect 71 0 70 0;
      #X connect 72 0 71 0;
      #X connect 73 0 70 0;
      #X connect 74 0 72 0;
1165  #X connect 74 0 73 0;
      #X connect 75 0 23 0;
```

```
#X connect 76 0 52 0;
#X connect 77 0 26 0;
#X connect 78 0 75 0;
1170 #X connect 78 1 21 0;
#X connect 79 0 22 0;
#X connect 79 1 27 0;
#X connect 79 1 28 0;
#X connect 79 1 83 1;
1175 #X connect 79 2 80 0;
#X connect 80 0 95 0;
#X connect 80 1 96 0;
#X connect 80 2 97 0;
#X connect 80 3 98 0;
1180 #X connect 80 4 99 0;
#X connect 80 5 81 0;
#X connect 81 0 7 5;
#X connect 82 0 52 0;
#X connect 83 0 26 0;
1185 #X connect 84 0 77 0;
#X connect 85 0 52 0;
#X connect 86 0 87 1;
#X connect 87 0 45 0;
#X connect 87 1 90 0;
1190 #X connect 87 2 108 0;
#X connect 87 3 104 0;
#X connect 87 4 53 0;
#X connect 88 0 87 2;
#X connect 89 0 87 3;
1195 #X connect 90 0 95 1;
#X connect 90 1 96 1;
#X connect 90 2 97 1;
#X connect 90 3 98 1;
#X connect 90 4 99 1;
1200 #X connect 90 5 81 1;
#X connect 90 6 100 0;
#X connect 91 0 90 0;
#X connect 92 0 91 0;
#X connect 92 0 48 1;
1205 #X connect 93 0 94 0;
#X connect 94 0 92 0;
#X connect 95 0 7 0;
#X connect 96 0 7 1;
#X connect 97 0 7 2;
1210 #X connect 98 0 7 3;
#X connect 99 0 7 4;
#X connect 100 0 9 0;
#X connect 100 1 11 0;
#X connect 100 2 14 0;
1215 #X connect 100 3 17 0;
#X connect 101 0 42 0;
#X connect 102 0 101 0;
#X connect 102 0 105 0;
#X connect 103 0 87 0;
1220 #X connect 103 0 111 0;
#X connect 105 0 37 2;
#X connect 108 0 106 0;
#X connect 108 0 105 1;
```

```

#X connect 108 1 107 0;
1225 #X connect 108 1 105 2;
#X connect 108 2 92 0;
#X connect 109 0 110 0;
#X connect 109 1 110 1;
#X connect 109 2 110 2;
1230 #X connect 110 0 112 0;
#X connect 111 0 113 0;
#X connect 112 0 113 0;
#X connect 113 0 115 0;
#X connect 113 1 114 0;
1235 #X connect 113 2 114 1;
#X connect 114 0 115 1;
#X connect 115 0 119 0;
#X connect 116 0 52 0;
#X connect 117 0 52 0;
1240 #X connect 118 0 116 0;
#X connect 118 1 117 0;

```

4 clap~.pd

```

#N canvas 108 34 777 632 10;
#X obj 228 19 noise~;
#X obj -66 450 *~;
#X obj -36 415 *~;
5 #X obj 168 266 *~ 0.8;
#X obj 13 446 *~ 0.2;
#X obj -52 164 del 30;
#X obj -27 196 del 20;
#X obj -10 231 del 15;
10 #X obj 9 269 del 10;
#X obj 64 233 *~ 0.1;
#X obj 118 203 bp~ 1300 10;
#X obj 286 203 bp~ 700 10;
#X obj 208 203 bp~ 2500 4;
15 #X obj -58 480 bp~ 2700 2;
#X obj 179 64 hip~ 300;
#X obj -207 333 *~;
#X obj -206 284 vline~;
#X obj -206 308 *~;
20 #X obj -116 330 *~;
#X obj -115 281 vline~;
#X obj -115 305 *~;
#X obj -112 235 bang;
#X obj 36 203 lop~ 10000;
25 #X obj -106 81 t b f;
#X obj -158 17 select 0;
#X obj -190 107 * 100;
#X obj -138 106 * 13;
#X obj -191 130 + 1000;
30 #X obj -138 130 + 1000;
#X obj -52 108 * 25;
#X obj 29 105 * 7;
#X obj 29 129 + 100;
#X obj -53 131 + 800;
35 #X obj -109 37 * 2;
#X obj -105 59 + 100;

```

```

#X obj -405 -1 noise~;
#X obj -268 261 *~ 0.8;
#X obj -372 228 *~ 0.1;
40 #X obj -354 199 bp~ 1300 10;
#X obj -186 199 bp~ 700 10;
#X obj -264 199 bp~ 2500 4;
#X obj -257 59 hip~ 300;
#X obj -436 199 lop~ 10000;
45 #X obj -270 452 *~;
#X obj -245 403 *~;
#X obj -191 448 *~ 0.2;
#X obj -262 482 bp~ 2700 2;
#X obj -49 4 noise~;
50 #X obj -159 -4 inlet;
#X obj -262 530 outlet~;
#X obj -262 503 expr~ tanh($v1);
#X obj -58 531 outlet~;
#X obj -58 504 expr~ tanh($v1);
55 #X msg -113 258 1.3 3 \, 0 100 3;
#X msg -214 257 1.3 1 \, 0 800 1;
#X connect 0 0 14 0;
#X connect 0 0 22 0;
#X connect 1 0 13 0;
60 #X connect 2 0 1 1;
#X connect 2 0 4 0;
#X connect 3 0 2 1;
#X connect 4 0 13 0;
#X connect 5 0 6 0;
65 #X connect 5 0 21 0;
#X connect 6 0 7 0;
#X connect 6 0 21 0;
#X connect 7 0 8 0;
#X connect 7 0 21 0;
70 #X connect 8 0 21 0;
#X connect 9 0 3 0;
#X connect 10 0 3 0;
#X connect 11 0 3 0;
#X connect 12 0 3 0;
75 #X connect 13 0 52 0;
#X connect 14 0 10 0;
#X connect 14 0 11 0;
#X connect 14 0 12 0;
#X connect 15 0 2 0;
80 #X connect 15 0 44 1;
#X connect 16 0 17 0;
#X connect 16 0 17 1;
#X connect 17 0 15 0;
#X connect 17 0 15 1;
85 #X connect 18 0 1 0;
#X connect 18 0 43 1;
#X connect 19 0 20 0;
#X connect 19 0 20 1;
#X connect 20 0 18 0;
90 #X connect 20 0 18 1;
#X connect 21 0 53 0;
#X connect 22 0 9 0;
#X connect 23 0 5 0;

```

```

#X connect 23 0 54 0;
95 #X connect 23 0 21 0;
#X connect 23 1 30 0;
#X connect 23 1 29 0;
#X connect 23 1 26 0;
#X connect 23 1 25 0;
100 #X connect 24 1 33 0;
#X connect 25 0 27 0;
#X connect 26 0 28 0;
#X connect 27 0 22 1;
#X connect 27 0 42 1;
105 #X connect 28 0 10 1;
#X connect 28 0 38 1;
#X connect 29 0 32 0;
#X connect 30 0 31 0;
#X connect 31 0 11 1;
110 #X connect 31 0 39 1;
#X connect 32 0 12 1;
#X connect 32 0 40 1;
#X connect 33 0 34 0;
#X connect 34 0 23 0;
115 #X connect 35 0 41 0;
#X connect 35 0 42 0;
#X connect 36 0 44 0;
#X connect 37 0 36 0;
#X connect 38 0 36 0;
120 #X connect 39 0 36 0;
#X connect 40 0 36 0;
#X connect 41 0 38 0;
#X connect 41 0 39 0;
#X connect 41 0 40 0;
125 #X connect 42 0 37 0;
#X connect 43 0 46 0;
#X connect 44 0 45 0;
#X connect 44 0 43 0;
#X connect 45 0 46 0;
130 #X connect 46 0 50 0;
#X connect 47 0 14 0;
#X connect 47 0 41 0;
#X connect 47 0 42 0;
#X connect 47 0 22 0;
135 #X connect 48 0 24 0;
#X connect 50 0 49 0;
#X connect 52 0 51 0;
#X connect 53 0 19 0;
#X connect 54 0 16 0;

```

5 COPYING

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
5 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program"

means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

180 5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
185 Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

190 6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
195 this License.

200 7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
205 license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

210 If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

215 It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
220 generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

225 This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

230 8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
235 the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

6 fmosc~.pd

```
#N canvas 0 0 450 300 10;
#X obj 19 271 outlet~;
#X obj 87 271 outlet~;
#X obj 84 2 inlet;
5 #X obj 84 25 route \$1;
#X obj 84 52 unpack f f f;
#X obj 19 189 osc~;
```

```

#X obj 126 233 + 1;
#X obj 88 233 *~ 0;
10 #X obj 18 233 *~ 0;
#X obj 53 232 + 1;
#X obj 126 212 * -1;
#X obj 20 109 osc~;
#X obj 20 140 *~ 0;
15 #X obj 20 168 +~ 0;
#X obj 163 157 * -300;
#X obj 163 179 + 600;
#X obj 125 176 mtof;
#X obj 123 114 int;
20 #X obj 124 134 * 12;
#X obj 124 95 + 6;
#X obj 123 73 * 5;
#X obj 295 8 inlet;
#X obj 126 154 + 0;
25 #X obj 179 110 t b f;
#X obj 181 133 mod 12;
#X connect 2 0 3 0;
#X connect 3 0 4 0;
#X connect 4 0 10 0;
30 #X connect 4 0 9 0;
#X connect 4 1 20 0;
#X connect 4 2 14 0;
#X connect 5 0 7 0;
#X connect 5 0 8 0;
35 #X connect 6 0 7 1;
#X connect 7 0 1 0;
#X connect 8 0 0 0;
#X connect 9 0 8 1;
#X connect 10 0 6 0;
40 #X connect 11 0 12 0;
#X connect 12 0 13 0;
#X connect 13 0 5 0;
#X connect 14 0 15 0;
#X connect 15 0 12 1;
45 #X connect 16 0 11 0;
#X connect 16 0 13 1;
#X connect 17 0 18 0;
#X connect 18 0 22 0;
#X connect 19 0 17 0;
50 #X connect 20 0 19 0;
#X connect 21 0 23 0;
#X connect 22 0 16 0;
#X connect 23 0 22 0;
#X connect 23 1 24 0;
55 #X connect 24 0 22 1;

```

7 Grid.lhs

polytopiary -- interaction in more dimensions
 Copyright (C) 2006, 2007 Claude Heiland-Allen

5 This program is free software; you can redistribute it and/or
 modify it under the terms of the GNU General Public License

as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

10 This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

15 You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

20 -----

Abstract data type.

25 -----

```

> module Grid (
>   Grid,
>   toGrid,
>   toList,
30  >   dims,
>   dim,
>   redim,
>   nest,
>   unnest,
35  >   join,
>   Grid.concat,
>   Grid.map,
>   Grid.zipWith,
>   for,
40  >   remap,
>   perspective,
>   outer,
>   inner,
>   innerl,
45  >   innerr,
>   rotation,

>   putGridPPM8Bit,

```

50 >) where

```

-----
> import Array
55  > import IO
> import MissingH.IO.HVIO
> import MissingH.IO.Binary

```

60 -----

Concrete data type.

```

65 -----
> data Grid a = G [Int] (Array Int a)
>   deriving (Show, Read, Eq)

70 -----

Grids are kinda numeric.

75 -----

> instance Num a => Num (Grid a)
>   where
>     fromInteger x = toGrid [] [fromInteger x]
80 >     negate      = Grid.map negate
>     (+)          = Grid.zipWith (+)
>     (*)          = Grid.zipWith (*)
>     abs          = Grid.map abs
>     signum       = Grid.map signum
85
> instance Fractional a => Fractional (Grid a)
>   where
>     fromRational x = toGrid [] [fromRational x]
>     recip          = Grid.map recip
90 >     (/)           = Grid.zipWith (/)

> instance Ord a => Ord (Grid a)
>   where
>     compare x y    = compare (toList x) (toList y)
95
> instance Real a => Real (Grid a)
>   where
>     toRational (G d a)
>       | length d == 1 && head d == 1 = toRational (a!0)
100 >       | otherwise                  = error "Grid.toRational: size != 1"

> instance Enum a => Enum (Grid a)
>   where
>     succ (G d a)
105 >       | length d == 1 && head d == 1 = toGrid [] [succ (a!0)]
>       | otherwise                  = error "Grid.succ: size != 1"
>     pred (G d a)
>       | length d == 1 && head d == 1 = toGrid [] [pred (a!0)]
>       | otherwise                  = error "Grid.pred: size != 1"
110 >     toEnum x                = toGrid [] [toEnum x]
>     fromEnum (G d a)
>       | length d == 1 && head d == 1 = fromEnum (a!0)
>       | otherwise                  = error "Grid.fromEnum: size != 1"

115 > instance Integral a => Integral (Grid a)
>   where
>     quot          = Grid.zipWith quot
>     rem           = Grid.zipWith rem
>     div           = Grid.zipWith div
120 >     mod           = Grid.zipWith mod

```

```

> quotRem x y = (Grid.zipWith quot x y, Grid.zipWith rem x y)
> divMod x y = (Grid.zipWith div x y, Grid.zipWith mod x y)
> toInteger (G d a)
>   | length d == 1 && head d == 1 = toInteger (a!0)
125 >   | otherwise = error "Grid.toInteger: size != 1"

> instance Floating a => Floating (Grid a)
>   where
>     pi = toGrid [] [pi]
130 >     exp = Grid.map exp
>     log = Grid.map log
>     sqrt = Grid.map sqrt
>     (**) = Grid.zipWith (**)
>     logBase = Grid.zipWith logBase
135 >     sin = Grid.map sin
>     cos = Grid.map cos
>     tan = Grid.map tan
>     asin = Grid.map asin
>     acos = Grid.map acos
140 >     atan = Grid.map atan
>     sinh = Grid.map sinh
>     cosh = Grid.map cosh
>     tanh = Grid.map tanh
>     asinh = Grid.map asinh
145 >     acosh = Grid.map acosh
>     atanh = Grid.map atanh

-- this stuff errors due to "inferred type is not general enough"
150 instance RealFrac a => RealFrac (Grid a)
    where
        ceiling = Grid.map ceiling
        properFraction g = (Grid.map fst h, Grid.map snd h)
155        where h = Grid.map properFraction g

-----

Validation.
160 -----

> assert :: (a -> String) -> (a->Bool) -> a -> a
> assert msg f x
165 >   | f x = x
>   | otherwise = error ("assert: " ++ (msg x))

> assert_dim :: Int -> Int
> assert_dim = assert (\x -> "dimension invalid: " ++ (show x)) (> 0)
170 > assert_dims :: [Int] -> [Int]
> assert_dims = filter (/= 0)

> assert_bounds :: Int -> Array Int a -> Array Int a
175 > assert_bounds n as = assert msg f as
>   where
>     msg x = "array bounds corrupt: " ++ (show (bounds x))

```

```

>     f a      = g (bounds a)
>     g (0,m) = n == m+1
180
>   assert_grid :: Grid a -> Grid a
>   assert_grid (G ds as) = G ods oas
>   where
>     ods = assert_dims ds
185 >     oas = assert_bounds (foldr (*) 1 ds) as

```

```

-----
190   Abstract constructor.

```

```

>   toGrid :: [Int] -> [a] -> Grid a
-----
195
>   toGrid ds as = assert_grid (G dims (array (0,len-1) cells))
>   where
>     cells = zip [0..(len-1)] (Prelude.concat (repeat as))
>     len   = foldr (*) 1 dims
200 >     dims  = assert_dims ds

```

```

-----
205   Export as a list.

```

```

>   toList :: Grid a -> [a]
-----
210
>   toList (G _ as) = [ as!i | i <- range (bounds as) ]

```

```

-----
215   Number of dimensions.

```

```

>   dims :: Grid a -> Int

```

```

220 -----
>   dims (G d _) = length d

```

```

225 -----
Dimensions.

```

```

>   dim :: Grid a -> Grid Int

```

```

230 -----
>   dim (G [] _) = toGrid [] [0]
>   dim (G d _) = toGrid [(length d)] d

```

235

Redimension a grid.

240 > redim :: [Int] -> Grid a -> Grid a

245 > redim ds g = toGrid ds (toList g)

Apply a function to every cell.

250 > map :: (a -> b) -> Grid a -> Grid b

255 > map f (G d x) = assert_grid (G d (array (bounds x)
> [(i, f (x!i)) | i <- range (bounds x)]))

260 Combine two grids, redimensioning the second to match the first.

> zipWith :: (a -> b -> c) -> Grid a -> Grid b -> Grid c

265

> zipWith f (G xd xa) (G yd ya) = toGrid xd (Prelude.zipWith f xl yl)
> where
> xl = toList (G xd xa)
270 > yl = Prelude.concat (repeat (toList (G yd ya)))

275 Nest grids at a given dimension index.

> nest :: Int -> Grid a -> Grid (Grid a)

280

> nest n (G ds as) = toGrid od gs
> where
> gs = [toGrid id [as!i | i <- [r .. r+si-1]]
> | r <- [0, si .. s-1]]
285 > si = foldr (*) 1 id
> s = foldr (*) 1 ds
> od = take m ds
> id = drop m ds
> m = assert (\x -> "cannot nest: " ++ (show x))
290 > (\x -> (0 <= x) && (x <= (length ds)))
> n

```

-----
295  Un-nest nested grids.

>  unnest :: Grid (Grid a) -> Grid a
-----
300

>  unnest (G ind ina) = toGrid outd outl
>    where
>      outd = ind ++ toList (dim (ina!0))
305 >      outl = Prelude.concat (toList (Grid.map toList (G ind ina)))
-----

310  Find coordinate corresponding to a subgrid.

>  coord :: [Int] -> [Int] -> Int
-----
315

>  coord ds xs = Prelude.foldr (+) 0 (Prelude.zipWith (*)
>                                     (Prelude.drop 1 (Prelude.scanr (*) 1 ds))
>                                     (xs ++ (repeat 0)))
-----
320

Reindex a grid.

325 >  reindex :: Grid Int -> Grid a -> Grid a
-----

>  reindex (G ld la) (G rd ra) = toGrid outd outl
330 >    where
>      outl      = [ ra!(coord rd [wa!i | i <- [r .. r+m-1]])
>                  | r <- [0, m .. s-1] ]
>      (G _ wa) = Grid.zipWith mod (G ld la) (dim (G rd ra))
>      s        = foldr (*) 1 ld
335 >      outd     = (Prelude.init ld) ++ subd
>      subd     = Prelude.drop m rd
>      m        = assert (\x -> "cannot reindex: " ++ (show x))
>                  (\x -> (0 < x) && (x <= (length rd)))
>                  (last ld)
340
-----

Join two grids at a given dimension.

345 >  join :: Int -> Grid a -> Grid a -> Grid a
-----

```

```

350 > join 0 (G ld la) (G rd ra)
>   | tail ld == tail rd = toGrid ([ head ld + head rd ]++tail ld)
>                               (toList (G ld la) ++ toList (G rd ra))
>   | otherwise          = error "Grid.join: mismatched tail dimensions"

355 > join n (G ld la) (G rd ra)
>   | take n ld == take n rd = unnest (Grid.zipWith (join 0)
>                                         (nest n (G ld la)) (nest n (G rd ra)))
>   | otherwise              = error "Grid.join: mismatched init dimensions"

```

360

Join a list of grids at a given dimension.

```

365 > concat :: Int -> [Grid a] -> Grid a

```

```

> concat n = foldr1 (join n)

```

370

Generate a grid of coordinates.

```

375 > for :: Grid a -> Grid Int -> Grid (a->a) -> Grid a

```

```

380 > lfor :: Int -> [a] -> [Int] -> [a->a] -> [[a]]
> lfor d f c s
>   | d == 1 = take (head c) l
>   | d > 1  = Prelude.concat (Prelude.zipWith op
>                               (lfor 1 [head f] [head c] [head s])
>                               (repeat (lfor (d-1) (tail f) (tail c) (tail s))))
385 >
>   where
>     op [x] y = Prelude.map (x:) y
>     l = [head f] : Prelude.map (Prelude.map (head s)) l

```

390

```

> for (G fromd froma) (G countd counta) (G stepd stepa)
>   | fromd == countd && countd == stepd && (length fromd) == 1 =
>     toGrid ((toList (G countd counta))++[head fromd]) (Prelude.concat
>                                                         (lfor
395 >                                     (head fromd)
>                                     (toList (G fromd froma))
>                                     (toList (G countd counta))
>                                     (toList (G stepd stepa))))
>   | otherwise = error "for: input size mismatched"

```

400

Remap a grid.

405

```
> remap :: (Grid Int -> Grid Int) -> Grid a -> Grid a
```

```
-----
410 > remap f g = reindex (f c) g
>   where
>     c = for (toGrid [dims g] [0])
>           (toGrid [dims g] (toList (dim g)))
>           (toGrid [dims g] [(+1)]))
415
```

```
-----
Perspective projection.
```

```
420 > perspective :: (Num a, Fractional a) => Grid a -> Grid a -> Grid a
```

```
-----
425 > perspective g z
>   | (last ds) > 1 = unnest $ Grid.zipWith o (nest ((length ds) - 1) g) z
>   | otherwise     = error "Grid.perspective: last dimension too small"
>   where
>     o (G [d] x) q = toGrid [d-1]
430 >                      [(x!i) / ((x!(d-1) / q) + 1) | i <- [0..d-2]]
>     G ds _       = g
```

```
-----
435 Outer product.
```

```
> outer :: (a -> b -> c) -> Grid a -> Grid b -> Grid c
```

```
-----
440
> outer o (G ld la) (G rd ra) = G od oa
>   where
>     oa = array ob [ (i*rs+j, o (la!i) (ra!j)) | i <- is, j <- js ]
445 >     ob = (0, os - 1)
>     os = foldr (*) 1 od
>     od = ld ++ rd
>     rs = foldr (*) 1 rd
>     is = range (bounds la)
450 >     js = range (bounds ra)
```

```
-----
455 Fold.
```

```
fold :: (b -> a -> b) -> b -> Grid a -> Grid b
```

```
-----
460 fold f s x = o x
      where
```

```
o gs      = foldl f s (toList gs)
```

465

```
-----
```

Inner product with arbitrary fold.

```
470 > inner :: (a -> b -> c) -> ([c] -> d) -> Grid a -> Grid b -> Grid d
```

```
-----
```

```
> inner o fold l r
475 > | (last ld) == (head rd) = G od (array ob oa)
> | otherwise              = error "Grid.inner: mismatched dimensions"
> where
>   oa      = [(oistep * i + okstep * k,
>               fold [ o
480 >                   (la!(listep * i + ljstep * j))
>                   (ra!(rjstep * j + rkstep * k))
>                   | j <- [0..jmax]
>                   ])
>               | i <- [0..limax]
485 >               , k <- [0..rkmax]
>               ]
>   ob      = (0,omax)
>   ld      = toList (dim l)
>   rd      = toList (dim r)
490 >   G _ la = l
>   G _ ra = r
>   od      = (init ld) ++ (tail rd)
>   omax     = (foldr (*) 1 od) - 1
>   oistep   = (foldr (*) 1 (tail rd))
495 >   okstep   = 1
>   limax    = (foldr (*) 1 (init ld)) - 1
>   listep   = last ld
>   ljstep   = 1
>   jmax     = (head rd) - 1
500 >   rjstep   = foldr (*) 1 (tail rd)
>   rkmax    = (foldr (*) 1 (tail rd)) - 1
>   rkstep   = 1
```

505

```
-----
```

Inner product with standard folds.

```
> innerl :: (a -> b -> c) -> (d -> c -> d) -> d -> Grid a -> Grid b -> Grid d
> innerr :: (a -> b -> c) -> (c -> d -> d) -> d -> Grid a -> Grid b -> Grid d
```

510

```
-----
```

```
> innerl o f s l r = inner o (foldl f s) l r
> innerr o f s l r = inner o (foldr f s) l r
```

515

```
-----
```

Rotation about all axes.

```

520 > rotation :: (Floating a) => Int -> [a] -> Grid a
> -----
525 > rotation d as
>   | length as == (d * (d-1)) `div` 2 = foldr (Grid.innerr (*) (+) 0)
>                                             (toGrid [d,d] ([1] ++ take d (\
>   ↪ repeat 0)))
>
>                                     rots
>   | otherwise = error "Grid.rotation: number of angles doesn't fit dimension"
530 >   where
>     rots = [ rot a i j
>              | (a,(i,j)) <- zip as [(i,j)
>              | i <- [0..(d-1)], j <- [i..(d-1)], i /= j ]
>              ]
535 >     rot a i j = Grid.map (r a i j) xy
>     xy = Grid.map toList (Grid.nest 2
>     (Grid.for (toGrid [2] [0]) (toGrid [2] [d]) (toGrid [2] [(+1)])))
>     r a i j [x,y]
>       | x == i && y == i = cos a
540 >       | x == i && y == j = sin a
>       | x == j && y == i = - (sin a)
>       | x == j && y == j = cos a
>       | x == y && x /= i && x /= j = 1
>       | otherwise       = 0
545

```

```

static Matrix<Number> rotation(unsigned int d, std::vector<Number> as) {
    assert(d > 1);
    Matrix<Number> r(Matrix<Number>::identity(d));
550    typename std::vector<Number>::iterator ai = as.begin();
    for (unsigned int i = 0; i < d; ++i) {
        for (unsigned int j = i + 1; j < d; ++j) {
            Matrix<Number> m(d, d);
            Number a = (*ai); ++ai;
555            for (unsigned int x = 0; x < d; ++x) {
                for (unsigned int y = 0; y < d; ++y) {
                    if (x == i && y == i) { m(x, y) = cos(a); }
                    else if (x == i && y == j) { m(x, y) = sin(a); }
                    else if (x == j && y == i) { m(x, y) = -sin(a); }
560                    else if (x == j && y == j) { m(x, y) = cos(a); }
                    else if (x == y) { m(x, y) = 1; }
                    else { m(x, y) = 0; }
                }
            }
565            r = r * m;
        }
    }
    return r;
}
570

```

PPM image output.

```

575     1. A "magic number" for identifying the file type. A ppm image's magic number↵

```

```

    ↪ is the two characters "P6".
2. Whitespace (blanks, TABs, CRs, LFs).
3. A width, formatted as ASCII characters in decimal.
4. Whitespace.
5. A height, again in ASCII decimal.
580 6. Whitespace.
7. The maximum color value (Maxval), again in ASCII decimal. Must be less ↪
    ↪ than 65536 and more than zero.
8. Newline or other single whitespace character.
9. A raster of Height rows, in order from top to bottom. Each row consists of ↪
    ↪ Width pixels, in order from left to right. Each pixel is a triplet of ↪
    ↪ red, green, and blue samples, in that order. Each sample is represented ↪
    ↪ in pure binary by either 1 or 2 bytes. If the Maxval is less than 256, ↪
    ↪ it is 1 byte. Otherwise, it is 2 bytes. The most significant byte is ↪
    ↪ first.

585 A row of an image is horizontal. A column is vertical. The pixels in the ↪
    ↪ image are square and contiguous.
10. In the raster, the sample values are "nonlinear." They are proportional to ↪
    ↪ the intensity of the ITU-R Recommendation BT.709 red, green, and blue ↪
    ↪ in the pixel, adjusted by the BT.709 gamma transfer function. (That ↪
    ↪ transfer function specifies a gamma number of 2.2 and has a linear ↪
    ↪ section for small intensities). A value of Maxval for all three samples ↪
    ↪ represents CIE D65 white and the most intense color in the color ↪
    ↪ universe of which the image is part (the color universe is all the ↪
    ↪ colors in all images to which this image might be compared).

    ITU-R Recommendation BT.709 is a renaming of the former CCIR ↪
    ↪ Recommendation 709. When CCIR was absorbed into its parent ↪
    ↪ organization, the ITU, ca. 2000, the standard was renamed. This ↪
    ↪ document once referred to the standard as CIE Rec. 709, but it isn't ↪
    ↪ clear now that CIE ever sponsored such a standard.

590 Note that another popular color space is the newer sRGB. A common ↪
    ↪ variation on PPM is to substitute this color space for the one ↪
    ↪ specified.
11. Note that a common variation on the PPM format is to have the sample ↪
    ↪ values be "linear," i.e. as specified above except without the gamma ↪
    ↪ adjustment. pnggamma takes such a PPM variant as input and produces a ↪
    ↪ true PPM as output.
12. Characters from a "#" to the next end-of-line, before the maxval line, are ↪
    ↪ comments and are ignored.

```

595

```
putGrid :: Handle -> (Handle -> a -> IO ()) -> Grid a -> IO [()]
```

600

```
-----
putGrid handle putter (G [h,w,3] a) = do
```

605

```
    PPM header
```

```
    hPutStr handle "P6 "
    hPutStr handle (show w)
```

```

        hPutStr handle " "
        hPutStr handle (show h)
610    hPutStr handle " "
        hPutStr handle (show 255)
        hPutChar handle '\n'

        pixel data
615    mapM (putter handle) [ a!i | i <- range (bounds a) ]

-----

620    > putGridPPM8Bit :: (HVIO h) => h -> Grid Char -> IO ()

-----

        > putGridPPM8Bit handle (G [h,w,3] a) = do
625    PPM header

        > vPutStr handle "P6\n# generated with gridskell\n"
        > vPutStr handle (show w)
630    > vPutStr handle " "
        > vPutStr handle (show h)
        > vPutStr handle " "
        > vPutStr handle (show 255)
        > vPutStr handle "\n"
635    pixel data

        > vPutStr handle [ a!i | i <- range (bounds a) ]
640
-----

-- EOF

```

8 hihat~.pd

```

#N canvas 0 0 325 344 10;
#X obj 17 18 inlet;
#X obj 17 43 select 0;
#X obj 66 113 vline~;
5  #X obj 66 135 *~;
   #X obj 66 155 *~;
   #X obj 114 148 noise~;
   #X obj 67 249 *~;
   #X obj 8 149 noise~;
10 #X obj 30 249 *~;
   #X obj 67 66 * 5;
   #X obj 97 172 *~;
   #X obj 7 173 *~;
   #X obj 96 128 noise~;
15 #X obj 97 216 hip~ 1500;
   #X obj 7 216 hip~ 1500;
   #X obj 97 196 hip~ 3000;
   #X obj 7 196 hip~ 3000;
   #X obj 11 279 expr~ tanh($v1) \; tanh($v2);

```

```

20  #X msg 67 87 1.3 3 \, 0 \$1 3;
    #X obj 12 320 outlet~;
    #X obj 118 321 outlet~;
    #X connect 0 0 1 0;
    #X connect 1 1 9 0;
25  #X connect 2 0 3 0;
    #X connect 2 0 3 1;
    #X connect 3 0 4 0;
    #X connect 3 0 4 1;
    #X connect 4 0 6 0;
30  #X connect 4 0 8 1;
    #X connect 5 0 10 1;
    #X connect 6 0 17 1;
    #X connect 7 0 11 0;
    #X connect 8 0 17 0;
35  #X connect 9 0 18 0;
    #X connect 10 0 15 0;
    #X connect 11 0 16 0;
    #X connect 12 0 10 0;
    #X connect 12 0 11 1;
40  #X connect 13 0 6 1;
    #X connect 14 0 8 0;
    #X connect 15 0 13 0;
    #X connect 16 0 14 0;
    #X connect 17 0 19 0;
45  #X connect 17 1 20 0;
    #X connect 18 0 2 0;

```

9 hsv2rgb.pd

```

    #N canvas 206 49 450 568 10;
    #X obj 187 84 route 0;
    #X msg 187 56 \$2 \$3 \$1;
    #X obj 187 409 f;
5  #X obj 231 111 unpack 0 0 0;
    #X text 250 71 svh;
    #X obj 187 28 inlet hsv;
    #X obj 187 520 outlet rgb;
    #X obj 401 149 * 6;
10  #X obj 380 196 % 6;
    #X obj 350 173 t f f;
    #X msg 350 149 0;
    #X text 406 196 i;
    #X obj 350 219 -;
15  #X obj 231 217 *;
    #X obj 231 178 - 1;
    #X obj 231 198 * -1;
    #X obj 263 330 *;
    #X obj 263 291 - 1;
20  #X obj 263 311 * -1;
    #X obj 263 269 *;
    #X obj 350 240 t f f;
    #X obj 380 262 - 1;
    #X obj 380 283 * -1;
25  #X obj 302 356 *;
    #X obj 302 317 - 1;
    #X obj 302 337 * -1;

```

```
#X obj 302 295 *;
#X obj 231 151 t f f f;
30 #X obj 241 392 pack 0 0 0 0;
#N canvas 387 255 646 315 select 0;
#X obj 290 37 inlet;
#X obj 16 247 outlet;
#X obj 92 104 t b l;
35 #X obj 125 128 unpack 0 0 0 0;
#X obj 89 43 inlet p q t v;
#X obj 92 128 i;
#X obj 92 151 select 0 1 2 3 4 5;
#X obj 16 223 pack 0 0 0;
40 #X obj 16 203 f;
#X obj 96 223 pack 0 0 0;
#X obj 96 203 f;
#X obj 176 223 pack 0 0 0;
#X obj 176 203 f;
45 #X obj 256 223 pack 0 0 0;
#X obj 256 203 f;
#X obj 336 223 pack 0 0 0;
#X obj 336 203 f;
#X obj 416 223 pack 0 0 0;
50 #X obj 416 203 f;
#X connect 0 0 5 1;
#X connect 2 0 5 0;
#X connect 2 1 3 0;
#X connect 3 0 7 2;
55 #X connect 3 0 9 2;
#X connect 3 0 12 1;
#X connect 3 0 14 1;
#X connect 3 0 15 1;
#X connect 3 0 17 1;
60 #X connect 3 1 10 1;
#X connect 3 1 13 1;
#X connect 3 1 17 2;
#X connect 3 2 7 1;
#X connect 3 2 11 2;
65 #X connect 3 2 16 1;
#X connect 3 3 8 1;
#X connect 3 3 9 1;
#X connect 3 3 11 1;
#X connect 3 3 13 2;
70 #X connect 3 3 15 2;
#X connect 3 3 18 1;
#X connect 4 0 2 0;
#X connect 5 0 6 0;
#X connect 6 0 8 0;
75 #X connect 6 1 10 0;
#X connect 6 2 12 0;
#X connect 6 3 14 0;
#X connect 6 4 16 0;
#X connect 6 5 18 0;
80 #X connect 7 0 1 0;
#X connect 8 0 7 0;
#X connect 9 0 1 0;
#X connect 10 0 9 0;
#X connect 11 0 1 0;
```

```
85  #X connect 12 0 11 0;
    #X connect 13 0 1 0;
    #X connect 14 0 13 0;
    #X connect 15 0 1 0;
    #X connect 16 0 15 0;
90  #X connect 17 0 1 0;
    #X connect 18 0 17 0;
    #X restore 241 416 pd select;
    #X msg 187 478 \$1 \$1 \$1;
    #X obj 187 450 clip 0 1;
95  #X obj 231 131 clip 0 1;
    #X obj 270 173 clip 0 1;
    #X obj 350 109 clip 0 1;
    #X obj 350 129 select 1;
    #X connect 0 0 2 0;
100 #X connect 0 1 3 0;
    #X connect 1 0 0 0;
    #X connect 2 0 31 0;
    #X connect 3 0 32 0;
    #X connect 3 1 33 0;
105 #X connect 3 2 34 0;
    #X connect 5 0 1 0;
    #X connect 7 0 9 0;
    #X connect 8 0 12 1;
    #X connect 8 0 29 1;
110 #X connect 9 0 12 0;
    #X connect 9 1 8 0;
    #X connect 10 0 9 0;
    #X connect 12 0 20 0;
    #X connect 13 0 28 0;
115 #X connect 14 0 15 0;
    #X connect 15 0 13 0;
    #X connect 16 0 28 1;
    #X connect 17 0 18 0;
    #X connect 18 0 16 0;
120 #X connect 19 0 17 0;
    #X connect 20 0 19 1;
    #X connect 20 1 21 0;
    #X connect 21 0 22 0;
    #X connect 22 0 26 1;
125 #X connect 23 0 28 2;
    #X connect 24 0 25 0;
    #X connect 25 0 23 0;
    #X connect 26 0 24 0;
    #X connect 27 0 14 0;
130 #X connect 27 1 19 0;
    #X connect 27 2 26 0;
    #X connect 28 0 29 0;
    #X connect 29 0 6 0;
    #X connect 30 0 6 0;
135 #X connect 31 0 30 0;
    #X connect 32 0 27 0;
    #X connect 33 0 13 1;
    #X connect 33 0 28 3;
    #X connect 33 0 16 1;
140 #X connect 33 0 23 1;
    #X connect 34 0 35 0;
```

```
#X connect 35 0 10 0;
#X connect 35 1 7 0;
```

10 Hyperspace.hs

```
-- d01234 -- d0->d1->d2->d3->d4
-- Copyright (C) 2007 Claude Heiland-Allen
--
--
5  -- This program is free software; you can redistribute it and/or
-- modify it under the terms of the GNU General Public License
-- as published by the Free Software Foundation; either version 2
-- of the License, or (at your option) any later version.
--
10 -- This program is distributed in the hope that it will be useful,
-- but WITHOUT ANY WARRANTY; without even the implied warranty of
-- MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
-- GNU General Public License for more details.
--
15 -- You should have received a copy of the GNU General Public License
-- along with this program; if not, write to the Free Software
-- Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, ↵
   ↵ USA.

20 module Hyperspace where

import Data.IORef
import PureData
import Grid

25 import Foreign.C.Types (CFloat)
import Foreign.Ptr (Ptr)

data HState = HState {
30   hColour      :: Ptr Symbol,
   hTranslate    :: Ptr Symbol,
   hRecalc       :: Ptr Symbol,
   hBang         :: Ptr Symbol,
   hCube         :: Grid CFloat,
35   hAngles      :: (CFloat, CFloat, CFloat, CFloat, CFloat, CFloat),
   hScales       :: (CFloat, CFloat, CFloat, CFloat),
   hCoords       :: [[CFloat]]
}

40 buildHypercube :: Int -> Grid CFloat
buildHypercube n = (Grid.for (toGrid [4] [-1]) (toGrid [4] [n]) (toGrid [4] [↵
   ↵ step])) )
   where
     step :: CFloat -> CFloat
     step = (+) (2/(fromInteger (toInteger n) - 1))

45 creator :: Creator
creator [AFloat n] = do
   colour    <- gensym "colour"
   translate <- gensym "translate"
50   recalc    <- gensym "recalc"
```

```

    bang      <- gensym "bang"
    stateref  <- newIORef HState{
        hColour    = colour ,
        hTranslate = translate ,
55      hRecalc     = recalc ,
        hBang      = bang ,
        hCube      = buildHypercube (fromIntegral (round n)),
        hAngles    = (0, 0, 0, 0, 0, 0),
        hScales    = (1, 1, 1, 1),
60      hCoords    = []
    }
    return (Just Instance{iInlets = 4, iOutlets = 1, iMethod = inlet stateref})

creator _ = do
65   return Nothing

-- inlets: getNext, recalc, angles, scales

70  inlet stateref outlet 0 _ _ = do
    state <- readIORef stateref
    writeIORef stateref state{hCoords = tail (hCoords state)}
    outlet 0 (hTranslate state) (coords state)
    return ()
75  where
    coords state = [AFloat x, AFloat y, AFloat z]
    where
        [x, y, z] = head (hCoords state)

80  inlet stateref outlet 1 _ _ = do
    state <- readIORef stateref
    writeIORef stateref state{hCoords = newcoords state}
    return ()
    where
85      newcoords state = toList (Grid.map toList (Grid.nest 4 (Grid.perspective
        ((Grid.innerr (*) (+) (0) ((hCube state) * (toGrid [4] [s0,s1,s2,s3]))
        (Grid.rotation 4 [a01,a02,a03,a12,a13,a23])) +
        (toGrid [4] [0,0,0,4])) 2)))
    where
90      (a01,a02,a12,a03,a13,a23) = hAngles state
        (s0,s1,s2,s3) = hScales state

inlet stateref outlet 2 _ [AFloat a01, AFloat a02, AFloat a03, AFloat a12, ↵
    ↵ AFloat a13, AFloat a23] = do
    state <- readIORef stateref
95  writeIORef stateref state{hAngles = (a01,a02,a12,a03,a13,a23)}
    return ()

inlet stateref outlet 3 _ [AFloat s0, AFloat s1, AFloat s2, AFloat s3] = do
    state <- readIORef stateref
100 writeIORef stateref state{hScales = (s0,s1,s2,s3)}
    return ()

inlet stateref outlet 2 _ _ = do
    return ()
105
inlet stateref outlet 3 _ _ = do

```

```
return ()
```

11 inkey.pd

```
#N canvas 0 0 1255 691 10;
#X obj 56 76 #out window \, title keyboard;
#X obj 56 36 loadbang;
#X msg 56 56 ( 64 256 3 # 0 );
5 #X obj 76 228 list prepend set;
#X obj 76 254 list trim;
#X msg 76 289;
#X connect 0 0 3 0;
#X connect 1 0 2 0;
10 #X connect 2 0 0 0;
#X connect 3 0 4 0;
#X connect 4 0 5 0;
```

12 kick~.pd

```
#N canvas 0 0 317 480 10;
#X obj 18 22 inlet;
#X obj 18 61 select 0;
#X obj 45 323 osc~;
5 #X obj 78 360 *~;
#X obj 73 439 outlet~;
#X obj 192 222 vline~;
#X obj 202 276 *~;
#X obj 201 250 *~;
10 #X obj 69 84 t f f;
#X obj 20 276 osc~;
#X obj 103 222 vline~;
#X obj 21 300 *~;
#X obj 107 254 *~;
15 #X obj 68 173 pack f f;
#X obj 20 253 *~ 0;
#X obj 79 383 expr~ tanh($v1);
#X obj 72 411 hip~ 5;
#X obj 89 284 *~;
20 #X obj 68 132 sqrt;
#X obj 68 152 sqrt;
#X msg 68 197 \ $1 5 \, 2.8 \ $2 5;
#X obj 114 145 * 2;
#X obj 69 107 * 8;
25 #X obj 56 278 *~;
#X msg 192 198 2 5 \, 0 \ $1 5;
#X connect 0 0 1 0;
#X connect 1 1 8 0;
#X connect 2 0 3 0;
30 #X connect 3 0 15 0;
#X connect 5 0 7 0;
#X connect 5 0 7 1;
#X connect 5 0 3 1;
#X connect 6 0 3 1;
35 #X connect 7 0 6 0;
#X connect 7 0 6 1;
#X connect 8 0 22 0;
#X connect 8 1 21 0;
```

```

#X connect 9 0 11 0;
40 #X connect 10 0 12 1;
#X connect 10 0 12 0;
#X connect 10 0 14 0;
#X connect 10 0 23 0;
#X connect 11 0 2 0;
45 #X connect 12 0 17 0;
#X connect 12 0 17 1;
#X connect 12 0 23 1;
#X connect 13 0 20 0;
#X connect 14 0 9 0;
50 #X connect 15 0 16 0;
#X connect 16 0 4 0;
#X connect 17 0 2 0;
#X connect 18 0 19 0;
#X connect 19 0 13 0;
55 #X connect 20 0 10 0;
#X connect 21 0 13 1;
#X connect 22 0 18 0;
#X connect 22 0 24 0;
#X connect 22 0 14 1;
60 #X connect 23 0 11 1;
#X connect 24 0 5 0;

```

13 Makefile

```

HOFLAGS = -fexcess-precision -fvia-C -optc-O2 -optc-mfpmath=sse -optc-msse2 -\
↳ optc-march=pentium-m

```

```

Grid.o: Grid.lhs
    ghc $(HOFLAGS) -o Grid.o -c Grid.lhs

```

14 mixtab-help.pd

```

#N canvas 0 0 450 300 10;
#X obj 43 83 mixtab 15;
#X obj 43 216 unpack f f;
#X floatatom 43 247 5 0 0 0 - - -;
5 #X floatatom 108 247 5 0 0 0 - - -;
#X connect 0 0 1 0;
#X connect 1 0 2 0;
#X connect 1 1 3 0;

```

15 mixtab.pd

```

#N canvas 0 0 450 300 10;
#X obj 101 101 cnv 15 248 123 empty empty mixtab 4 10 1 12 -123526
-241291 0;
#X obj 105 120 cnv 15 240 100 empty empty empty 20 12 0 14 -260818
5 -260818 0;
#X floatatom 108 135 3 0 63 0 - #0-66-r #0-66-s;
#X floatatom 108 149 3 0 127 0 - #0-40-r #0-40-s;
#X floatatom 108 163 3 0 127 0 - #0-32-r #0-32-s;
#X floatatom 108 177 3 0 127 0 - #0-24-r #0-24-s;
10 #X obj 108 124 hradio 11 1 0 2 \ $0-66s-s \ $0-66s-r empty 0 -8 0 10
-260818 -1 -1 0;

```

```

#X obj 108 191 hradio 11 1 0 2 \ $0-16s-s \ $0-16s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X floatatom 108 202 3 1 127 0 - #0-16-r #0-16-s;
15 #X floatatom 131 135 3 0 63 0 - #0-67-r #0-67-s;
#X floatatom 131 149 3 0 127 0 - #0-41-r #0-41-s;
#X floatatom 131 163 3 0 127 0 - #0-33-r #0-33-s;
#X floatatom 131 177 3 0 127 0 - #0-25-r #0-25-s;
#X obj 131 124 hradio 11 1 0 2 \ $0-67s-s \ $0-67s-r empty 0 -8 0 10
20 -260818 -1 -1 0;
#X obj 131 191 hradio 11 1 0 2 \ $0-17s-s \ $0-17s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X floatatom 131 202 3 1 127 0 - #0-17-r #0-17-s;
#X floatatom 154 135 3 0 63 0 - #0-68-r #0-68-s;
25 #X floatatom 154 149 3 0 127 0 - #0-42-r #0-42-s;
#X floatatom 154 163 3 0 127 0 - #0-34-r #0-34-s;
#X floatatom 154 177 3 0 127 0 - #0-26-r #0-26-s;
#X obj 154 124 hradio 11 1 0 2 \ $0-68s-s \ $0-68s-r empty 0 -8 0 10
-260818 -1 -1 0;
30 #X obj 154 191 hradio 11 1 0 2 \ $0-18s-s \ $0-18s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X floatatom 154 202 3 1 127 0 - #0-18-r #0-18-s;
#X floatatom 177 135 3 0 63 0 - #0-69-r #0-69-s;
#X floatatom 177 149 3 0 127 0 - #0-43-r #0-43-s;
35 #X floatatom 177 163 3 0 127 0 - #0-35-r #0-35-s;
#X floatatom 177 177 3 0 127 0 - #0-27-r #0-27-s;
#X obj 177 124 hradio 11 1 0 2 \ $0-69s-s \ $0-69s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X obj 177 191 hradio 11 1 0 2 \ $0-19s-s \ $0-19s-r empty 0 -8 0 10
40 -260818 -1 -1 0;
#X floatatom 177 202 3 1 127 0 - #0-19-r #0-19-s;
#X floatatom 200 135 3 0 63 0 - #0-70-r #0-70-s;
#X floatatom 200 149 3 0 127 0 - #0-44-r #0-44-s;
#X floatatom 200 163 3 0 127 0 - #0-36-r #0-36-s;
45 #X floatatom 200 177 3 0 127 0 - #0-28-r #0-28-s;
#X obj 200 124 hradio 11 1 0 2 \ $0-70s-s \ $0-70s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X obj 200 191 hradio 11 1 0 2 \ $0-20s-s \ $0-20s-r empty 0 -8 0 10
-260818 -1 -1 0;
50 #X floatatom 200 202 3 1 127 0 - #0-20-r #0-20-s;
#X floatatom 223 135 3 0 63 0 - #0-71-r #0-71-s;
#X floatatom 223 149 3 0 127 0 - #0-45-r #0-45-s;
#X floatatom 223 163 3 0 127 0 - #0-37-r #0-37-s;
#X floatatom 223 177 3 0 127 0 - #0-29-r #0-29-s;
55 #X obj 223 124 hradio 11 1 0 2 \ $0-71s-s \ $0-71s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X obj 223 191 hradio 11 1 0 2 \ $0-21s-s \ $0-21s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X floatatom 223 202 3 1 127 0 - #0-21-r #0-21-s;
60 #X floatatom 246 135 3 0 63 0 - #0-72-r #0-72-s;
#X floatatom 246 149 3 0 127 0 - #0-46-r #0-46-s;
#X floatatom 246 163 3 0 127 0 - #0-38-r #0-38-s;
#X floatatom 246 177 3 0 127 0 - #0-30-r #0-30-s;
#X obj 246 124 hradio 11 1 0 2 \ $0-72s-s \ $0-72s-r empty 0 -8 0 10
65 -260818 -1 -1 0;
#X obj 246 191 hradio 11 1 0 2 \ $0-22s-s \ $0-22s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X floatatom 246 202 3 1 127 0 - #0-22-r #0-22-s;

```

```

#X floatatom 269 135 3 0 63 0 - #0-73-r #0-73-s;
70 #X floatatom 269 149 3 0 127 0 - #0-47-r #0-47-s;
#X floatatom 269 163 3 0 127 0 - #0-39-r #0-39-s;
#X floatatom 269 177 3 0 127 0 - #0-31-r #0-31-s;
#X obj 269 124 hradio 11 1 0 2 \ $0-73s-s \ $0-73s-r empty 0 -8 0 10
-260818 -1 -1 0;
75 #X obj 269 191 hradio 11 1 0 2 \ $0-23s-s \ $0-23s-r empty 0 -8 0 10
-260818 -1 -1 0;
#X floatatom 269 202 3 1 127 0 - #0-23-r #0-23-s;
#X floatatom 296 135 3 0 127 0 - #0-74-r #0-74-s;
#X floatatom 296 149 3 0 127 0 - #0-80-r #0-80-s;
80 #X floatatom 296 163 3 0 127 0 - #0-78-r #0-78-s;
#X floatatom 296 177 3 0 127 0 - #0-76-r #0-76-s;
#X floatatom 296 202 3 0 127 0 - #0-7-r #0-7-s;
#X floatatom 319 135 3 0 127 0 - #0-75-r #0-75-s;
#X floatatom 319 149 3 0 127 0 - #0-81-r #0-81-s;
85 #X floatatom 319 163 3 0 127 0 - #0-79-r #0-79-s;
#X floatatom 319 177 3 0 127 0 - #0-77-r #0-77-s;
#X floatatom 319 202 3 0 127 0 - #0-48-r #0-48-s;
#N canvas 475 0 450 604 \ $0-guts 0;
#X obj 19 111 outlet;
90 #X obj 253 156 s \ $0-value-r;
#X obj 318 135 s \ $0-controller-r;
#X obj 260 3 ctlin;
#X obj 253 108 unpack f f;
#X obj 224 41 pack f f;
95 #X obj 253 73 spigot;
#X obj 290 44 == \ $1;
#X obj 18 156 unpack f f;
#X obj 69 542 send;
#X obj 92 520 makefilename \ $0-%d-r;
100 #X obj 99 202 select 16 17 18 19 20 21 22 23;
#X obj 18 326 shunt 3;
#X msg 99 219 0;
#X obj 99 242 select 66 67 68 69 70 71 72 73;
#X msg 99 259 1;
105 #X msg 99 288 2;
#X obj 69 181 t f f;
#X obj 18 205 pack f f;
#X obj 18 432 unpack f f;
#X obj 18 461 select 0;
110 #X obj 18 542 send;
#X msg 18 496 1;
#X obj 103 486 makefilename \ $0-%ds-r;
#X msg 47 496 0;
#X obj 254 339 send;
115 #X obj 277 317 makefilename \ $0-%d-r;
#X obj 253 289 unpack f f;
#X obj 169 462 send;
#X obj 192 440 makefilename \ $0-%d-r;
#X obj 118 352 unpack f f;
120 #X obj 118 462 send;
#X obj 203 406 makefilename \ $0-%ds-r;
#X obj 118 424 div 64;
#X obj 128 404 mod 64;
#X connect 3 0 5 0;
125 #X connect 3 1 5 1;

```

```
#X connect 3 2 7 0;
#X connect 4 0 1 0;
#X connect 4 1 2 0;
#X connect 5 0 6 0;
130 #X connect 6 0 4 0;
#X connect 6 0 8 0;
#X connect 6 0 0 0;
#X connect 7 0 6 1;
#X connect 8 0 18 0;
135 #X connect 8 1 17 0;
#X connect 10 0 9 1;
#X connect 11 0 13 0;
#X connect 11 1 13 0;
#X connect 11 2 13 0;
140 #X connect 11 3 13 0;
#X connect 11 4 13 0;
#X connect 11 5 13 0;
#X connect 11 6 13 0;
#X connect 11 7 13 0;
145 #X connect 11 8 14 0;
#X connect 12 0 19 0;
#X connect 12 1 30 0;
#X connect 12 2 27 0;
#X connect 13 0 12 1;
150 #X connect 14 0 15 0;
#X connect 14 1 15 0;
#X connect 14 2 15 0;
#X connect 14 3 15 0;
#X connect 14 4 15 0;
155 #X connect 14 5 15 0;
#X connect 14 6 15 0;
#X connect 14 7 15 0;
#X connect 14 8 16 0;
#X connect 15 0 12 1;
160 #X connect 16 0 12 1;
#X connect 17 0 18 1;
#X connect 17 1 11 0;
#X connect 18 0 12 0;
#X connect 19 0 20 0;
165 #X connect 19 1 10 0;
#X connect 19 1 23 0;
#X connect 20 0 22 0;
#X connect 20 1 9 0;
#X connect 20 1 24 0;
170 #X connect 22 0 21 0;
#X connect 23 0 21 1;
#X connect 24 0 21 0;
#X connect 26 0 25 1;
#X connect 27 0 25 0;
175 #X connect 27 1 26 0;
#X connect 29 0 28 1;
#X connect 30 0 33 0;
#X connect 30 0 34 0;
#X connect 30 1 29 0;
180 #X connect 30 1 32 0;
#X connect 32 0 31 1;
#X connect 33 0 31 0;
```

```

#X connect 34 0 28 0;
#X restore 12 21 pd \${0-guts};
185 #X obj 12 49 outlet;
#X obj 299 104 cnv 4 46 14 empty empty empty 20 12 0 14 -241291 -241291
0;
#X floatatom 299 104 3 0 127 0 - #0-controller-r #0-controller-s;
#X floatatom 322 104 3 0 127 0 - #0-value-r #0-value-s;
190 #X text 16 71 note: only MIDI -> outlet \, nbx's are display only...
;
#X connect 68 0 69 0;
#X coords 0 -1 1 1 250 125 2 100 100;

```

16 nreceive-abs.pd

```

#N canvas 0 0 313 125 10;
#X obj 14 18 r \${1};
#X obj 14 38 list prepend \${2};
#X obj 14 58 list trim;
5 #X obj 14 78 s \${3};
#X connect 0 0 1 0;
#X connect 1 0 2 0;
#X connect 2 0 3 0;

```

17 nreceive-help.pd

```

#N canvas 0 0 450 300 10;
#X obj 26 47 s \${0-0-r};
#X obj 96 47 s \${0-1-r};
#X obj 166 47 s \${0-2-r};
5 #X floatatom 26 26 5 0 0 0 - - -;
#X floatatom 96 26 5 0 0 0 - - -;
#X floatatom 166 26 5 0 0 0 - - -;
#X floatatom 36 225 5 0 0 0 - - -;
#X floatatom 100 225 5 0 0 0 - - -;
10 #X obj 76 182 route 0 1 2;
#X floatatom 160 225 5 0 0 0 - - -;
#X obj 76 141 nreceive \${0-%d-r 0 3};
#X text 12 122 arguments are: sender-pattern start-number count;
#X text 13 70 really you would use the GUI send-name field \, but this
15 is a help patch;
#X connect 3 0 0 0;
#X connect 4 0 1 0;
#X connect 5 0 2 0;
#X connect 8 0 6 0;
20 #X connect 8 1 7 0;
#X connect 8 2 9 0;
#X connect 10 0 8 0;

```

18 nreceive.pd

```

#N canvas 0 0 368 144 10;
#N canvas 0 0 450 300 \${0-receivers 0};
#X restore 12 13 pd \${0-receivers};
#X obj 12 54 outlet;
5 #X obj 12 34 r \${0-from-receivers};
#N canvas 223 157 443 424 \${0-constructor 0};

```

```

#X obj 13 16 inlet;
#X obj 13 168 f \$2;
#X obj 13 79 f \$3;
10 #X obj 13 105 until;
#X obj 90 178 + 1;
#X obj 13 222 makefilename \$1;
#X obj 13 48 t b b b;
#X obj 95 81 f \$0;
15 #X obj 95 107 makefilename pd-%d-receivers;
#X obj 13 324 send;
#X obj 13 192 t f f f;
#X obj 13 257 pack s f s;
#X obj 78 135 symbol \$0-from-receivers;
20 #X msg 13 292 obj 10 10 nreceive-abs \$1 \$2 \$3;
#X connect 0 0 6 0;
#X connect 1 0 10 0;
#X connect 2 0 3 0;
#X connect 3 0 1 0;
25 #X connect 4 0 1 1;
#X connect 5 0 11 0;
#X connect 6 0 2 0;
#X connect 6 1 12 0;
#X connect 6 2 7 0;
30 #X connect 7 0 8 0;
#X connect 8 0 9 1;
#X connect 10 0 5 0;
#X connect 10 1 11 1;
#X connect 10 2 4 0;
35 #X connect 11 0 13 0;
#X connect 12 0 11 2;
#X connect 13 0 9 0;
#X restore 13 114 pd \$0-processor;
#X obj 13 88 loadbang;
40 #X connect 2 0 1 0;
#X connect 4 0 3 0;

```

19 Polytopiary.lhs

polytopiary -- interaction in more dimensions
 Copyright (C) 2006, 2007 Claude Heiland-Allen

- 5 This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
- 10 This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
- 15 You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

```

20 > module Main where

    > import Graphics.Rendering.OpenGL
    > import Graphics.UI.GLUT
    > import Data.IORef
25 > import Data.Char ( chr , toLower )
    > import Data.Word
    > import System.Exit ( exitWith , ExitCode(ExitSuccess) )
    > import Foreign.Ptr
    > import Foreign.Marshal.Array
30 > import IO
    > import System.IO ( openBinaryFile , hClose )
    > import MissingH.IO.Binary
    > import Foreign.C.Types
    > import Foreign.C.String
35 > import Foreign.Marshal.Alloc

    > import Grid

    > data PState = PState {
40 >   pAngles    :: (GLfloat , GLfloat , GLfloat , GLfloat , GLfloat , GLfloat) ,
    >   pSpeeds    :: (GLfloat , GLfloat , GLfloat , GLfloat , GLfloat , GLfloat) ,
    >   pWidth     :: GLsizei ,
    >   pHeight    :: GLsizei ,
    >   pSnap      :: Int ,
45 >   pPixels     :: Ptr CChar ,
    >   pHandle     :: PPMWriterHandle
    > }

    > buildHypercube :: Int -> Grid GLfloat
50 > buildHypercube n = (Grid.for (toGrid [4] [-1]) (toGrid [4] [n]) (toGrid [4] [↯
    ↵ step])) )
    >   where
    >     step :: GLfloat -> GLfloat
    >     step = (+) (2/(fromInteger (toInteger n) - 1))

55 > hd = 4 :: Integer
    > hypercube = buildHypercube (fromInteger hd)

    > display stateref = do
    >   state <- get stateref
60 >   let (a,b,c,d,e,f) = pAngles state
    >   let s              = pSnap    state
    >   let w              = pWidth   state
    >   let h              = pHeight  state
    >   let p              = pPixels  state
65 >   let handle          = pHandle  state
    >   clear [ColorBuffer , DepthBuffer ]
    >   preservingMatrix $ do

    >     scale 2 2 (2::GLfloat)
70 >     mapM_ (\([x,y,z] ,(cr ,cg ,cb)) -> preservingMatrix $ do

    >       materialSpecular Front $= Color4 cr cg cb 1
    >       materialAmbient  Front $= Color4 cg cb cr 1
    >       materialDiffuse  Front $= Color4 cb cr cg 1
75 >       materialShininess Front $= 50

```

```

>         translate $ Vector3 x y z
>         renderQuadric (QuadricStyle (Just Smooth)
>                                     NoTextureCoordinates Outside
>                                     FillStyle) (Sphere 0.03 12 12)) $
80 >         zip
>         (toList (Grid.map toList (Grid.nest 4 (Grid.perspective
>         ((Grid.innerr (*) (+) (0) hypercube
>         (Grid.rotation 4 [a,b,c,d,e,f])) + (toGrid [4] [0,0,0,4])) 2))))
>         (Prelude.concat (repeat
85 >         [(fromInteger cr/fromInteger hd,
>         fromInteger cg/fromInteger hd,
>         fromInteger cb/fromInteger hd) | cr <- [1..hd], cg <- [1..hd], cb <-
>         ↵ <- [1..hd]]))
>         swapBuffers

90 > idle stateref = do
>     state <- get stateref
>     let ( a, b, c, d, e, f) = pAngles state
>     let (sa,sb,sc,sd,se,sf) = pSpeeds state
>     stateref $= state{pAngles =
95 >         (a + 2*pi*sa/3600, b + 2*pi*sb/3600, c + 2*pi*sc/3600,
>         d + 2*pi*sd/3600, e + 2*pi*se/3600, f + 2*pi*sf/3600)}
>     postRedisplay Nothing

```

20 psykaw~.pd

```

#N canvas 121 156 562 444 10;
#X obj 247 414 outlet~;
#X obj 354 412 outlet~;
#X obj 23 14 inlet;
5 #X obj 157 16 inlet;
#X obj 23 41 route \$1;
#X obj 23 67 unpack f f f;
#X obj 62 89 + 10;
#X obj 23 114 * 5;
10 #X obj 156 61 select 0;
#X obj 42 158 pack f f f;
#X obj 120 158 pack f f;
#X obj 119 242 vline~;
#X obj 111 266 *~;
15 #X obj 111 287 *~;
#X msg 150 198 0 \$2;
#X obj 189 206 phasor~;
#X obj 249 206 phasor~;
#X obj 309 206 phasor~;
20 #X obj 369 207 phasor~;
#X obj 429 206 phasor~;
#X obj 308 151 *~ 0.5;
#X obj 306 229 +~;
#X obj 324 117 *~ 2;
25 #X obj 242 79 mtof~;
#X obj 248 336 *~;
#X obj 354 341 *~;
#X obj 370 186 +~ 0.1;
#X obj 250 185 -~ 0.1;
30 #X obj 190 185 +~ 0.2;
#X obj 430 185 -~ 0.2;

```

```
#X obj 248 284 -~ 2;
#X obj 369 284 -~ 2;
#X obj 371 309 hip~ 5;
35 #X obj 248 304 hip~ 5;
#X obj 248 374 cos~;
#X obj 248 355 +~ 0.25;
#X obj 353 381 cos~;
#X obj 353 362 +~ 0.25;
40 #X obj 23 89 + 5;
#X obj 62 114 * 10;
#X obj 102 89 + 5;
#X obj 102 114 * 25;
#X obj 243 57 lop~ \$1;
45 #X msg 36 197 1 \$2 \, 0.8 \$3 \$2;
#X obj 157 39 change -1;
#X connect 2 0 4 0;
#X connect 3 0 44 0;
#X connect 4 0 5 0;
50 #X connect 5 0 38 0;
#X connect 5 1 6 0;
#X connect 5 1 27 1;
#X connect 5 1 26 1;
#X connect 5 2 29 1;
55 #X connect 5 2 28 1;
#X connect 5 2 40 0;
#X connect 6 0 39 0;
#X connect 7 0 9 1;
#X connect 8 0 10 0;
60 #X connect 8 1 42 0;
#X connect 8 1 9 0;
#X connect 9 0 43 0;
#X connect 10 0 14 0;
#X connect 11 0 12 1;
65 #X connect 11 0 12 0;
#X connect 12 0 13 0;
#X connect 12 0 13 1;
#X connect 13 0 24 1;
#X connect 13 0 25 0;
70 #X connect 14 0 11 0;
#X connect 15 0 30 0;
#X connect 16 0 30 0;
#X connect 17 0 21 0;
#X connect 17 0 21 1;
75 #X connect 18 0 31 0;
#X connect 19 0 31 0;
#X connect 20 0 26 0;
#X connect 20 0 27 0;
#X connect 21 0 30 0;
80 #X connect 21 0 31 0;
#X connect 22 0 28 0;
#X connect 22 0 29 0;
#X connect 23 0 22 0;
#X connect 23 0 20 0;
85 #X connect 23 0 17 0;
#X connect 24 0 35 0;
#X connect 25 0 37 0;
#X connect 26 0 18 0;
```

```

#X connect 27 0 16 0;
90 #X connect 28 0 15 0;
#X connect 29 0 19 0;
#X connect 30 0 33 0;
#X connect 31 0 32 0;
#X connect 32 0 25 1;
95 #X connect 33 0 24 0;
#X connect 34 0 0 0;
#X connect 35 0 34 0;
#X connect 36 0 1 0;
#X connect 37 0 36 0;
100 #X connect 38 0 7 0;
#X connect 39 0 9 2;
#X connect 40 0 41 0;
#X connect 41 0 10 1;
#X connect 42 0 23 0;
105 #X connect 43 0 11 0;
#X connect 44 0 8 0;

```

21 README

work in progress...

22 rotor.pd

```

#N canvas 0 0 585 606 10;
#X obj 101 101 cnv 15 48 48 empty empty empty 20 12 0 14 -225280 -1109
0;
#X floatatom 103 104 2 1 99 0 - #0-top-r #0-top-s;
5 #X floatatom 132 104 2 1 99 0 / #0-bot-r #0-bot-s;
#X floatatom 132 118 2 0 99 0 * #0-mul-r #0-mul-s;
#X floatatom 132 132 2 0 99 0 + #0-add-r #0-add-s;
#X obj 103 120 vsl 16 23 0 1 0 0 \ $0-pos-s \ $0-pos-r empty 0 -9 0 10
-225280 -1109 -1109 2200 1;
10 #X obj 251 14 r \ $0-top-s;
#X obj 291 34 r \ $0-bot-s;
#X obj 260 57 t b f;
#X obj 251 111 unpack f f;
#X obj 251 139 /;
15 #X obj 251 175 * 111;
#X obj 251 210 swap 1000;
#X obj 251 238 /;
#X obj 251 84 pack 1 1;
#X obj 251 265 phasor~;
20 #X obj 395 70 inlet;
#X obj 206 444 snapshot~;
#X obj 251 325 *~ 0;
#X obj 274 304 / 24.75;
#X obj 274 284 r \ $0-mul-s;
25 #X obj 250 388 +~ 0;
#X obj 273 345 r \ $0-add-s;
#X obj 273 366 / 99;
#X obj 250 411 wrap~;
#X obj 206 481 * 6.28319;
30 #X obj 206 161 inlet;
#X obj 287 481 s \ $0-pos-r;
#X obj 206 507 outlet;

```

```

#X obj 395 93 route top bot bpm mul add;
#X obj 395 125 s \${0-top-r;
35 #X obj 281 153 r \${0-bpm-r;
#X obj 405 155 s \${0-bot-r;
#X obj 415 185 s \${0-bpm-r;
#X obj 425 215 s \${0-mul-r;
#X obj 435 245 s \${0-add-r;
40 #X connect 6 0 14 0;
#X connect 7 0 8 0;
#X connect 8 0 14 0;
#X connect 8 1 14 1;
#X connect 9 0 10 0;
45 #X connect 9 1 10 1;
#X connect 10 0 11 0;
#X connect 11 0 12 0;
#X connect 12 0 13 0;
#X connect 12 1 13 1;
50 #X connect 13 0 15 0;
#X connect 14 0 9 0;
#X connect 15 0 18 0;
#X connect 16 0 29 0;
#X connect 17 0 25 0;
55 #X connect 17 0 27 0;
#X connect 18 0 21 0;
#X connect 19 0 18 1;
#X connect 20 0 19 0;
#X connect 21 0 24 0;
60 #X connect 22 0 23 0;
#X connect 23 0 21 1;
#X connect 24 0 17 0;
#X connect 25 0 28 0;
#X connect 26 0 17 0;
65 #X connect 29 0 30 0;
#X connect 29 1 32 0;
#X connect 29 2 33 0;
#X connect 29 3 34 0;
#X connect 29 4 35 0;
70 #X connect 31 0 11 1;
#X coords 0 -1 1 1 50 50 1 100 100;

```

23 seq256-gui-help.pd

```

#N canvas 0 0 656 415 10;
#X obj 71 65 seq256-gui;
#X obj 71 345 unpack f f;
#X floatatom 71 375 5 0 0 0 - - -;
5 #X floatatom 136 375 5 0 0 0 - - -;
#X obj 206 345 s \${0-table;
#N canvas 0 0 450 300 (subpatch) 0;
#X array \${0-table 256 float 2;
#X coords 0 99 256 0 256 256 1;
10 #X restore 368 64 graph;
#X connect 0 0 1 0;
#X connect 0 0 4 0;
#X connect 1 0 2 0;
#X connect 1 1 3 0;

```

24 seq256-gui.pd

```

#N canvas 0 0 450 300 10;
#X obj 11 11 cnv 15 280 250 \ $0-canvas-s \ $0-canvas-r empty 20 12 0
14 -257472 -66577 0;
#X floatatom 19 19 2 0 99 0 - #0-0-r #0-0-s;
5 #X floatatom 35 19 2 0 99 0 - #0-1-r #0-1-s;
#X floatatom 51 19 2 0 99 0 - #0-2-r #0-2-s;
#X floatatom 67 19 2 0 99 0 - #0-3-r #0-3-s;
#X floatatom 86 19 2 0 99 0 - #0-4-r #0-4-s;
#X floatatom 102 19 2 0 99 0 - #0-5-r #0-5-s;
10 #X floatatom 118 19 2 0 99 0 - #0-6-r #0-6-s;
#X floatatom 134 19 2 0 99 0 - #0-7-r #0-7-s;
#X floatatom 153 19 2 0 99 0 - #0-8-r #0-8-s;
#X floatatom 169 19 2 0 99 0 - #0-9-r #0-9-s;
#X floatatom 185 19 2 0 99 0 - #0-10-r #0-10-s;
15 #X floatatom 201 19 2 0 99 0 - #0-11-r #0-11-s;
#X floatatom 220 19 2 0 99 0 - #0-12-r #0-12-s;
#X floatatom 236 19 2 0 99 0 - #0-13-r #0-13-s;
#X floatatom 252 19 2 0 99 0 - #0-14-r #0-14-s;
#X floatatom 268 19 2 0 99 0 - #0-15-r #0-15-s;
20 #X floatatom 19 33 2 0 99 0 - #0-16-r #0-16-s;
#X floatatom 35 33 2 0 99 0 - #0-17-r #0-17-s;
#X floatatom 51 33 2 0 99 0 - #0-18-r #0-18-s;
#X floatatom 67 33 2 0 99 0 - #0-19-r #0-19-s;
#X floatatom 86 33 2 0 99 0 - #0-20-r #0-20-s;
25 #X floatatom 102 33 2 0 99 0 - #0-21-r #0-21-s;
#X floatatom 118 33 2 0 99 0 - #0-22-r #0-22-s;
#X floatatom 134 33 2 0 99 0 - #0-23-r #0-23-s;
#X floatatom 153 33 2 0 99 0 - #0-24-r #0-24-s;
#X floatatom 169 33 2 0 99 0 - #0-25-r #0-25-s;
30 #X floatatom 185 33 2 0 99 0 - #0-26-r #0-26-s;
#X floatatom 201 33 2 0 99 0 - #0-27-r #0-27-s;
#X floatatom 220 33 2 0 99 0 - #0-28-r #0-28-s;
#X floatatom 236 33 2 0 99 0 - #0-29-r #0-29-s;
#X floatatom 252 33 2 0 99 0 - #0-30-r #0-30-s;
35 #X floatatom 268 33 2 0 99 0 - #0-31-r #0-31-s;
#X floatatom 19 47 2 0 99 0 - #0-32-r #0-32-s;
#X floatatom 35 47 2 0 99 0 - #0-33-r #0-33-s;
#X floatatom 51 47 2 0 99 0 - #0-34-r #0-34-s;
#X floatatom 67 47 2 0 99 0 - #0-35-r #0-35-s;
40 #X floatatom 86 47 2 0 99 0 - #0-36-r #0-36-s;
#X floatatom 102 47 2 0 99 0 - #0-37-r #0-37-s;
#X floatatom 118 47 2 0 99 0 - #0-38-r #0-38-s;
#X floatatom 134 47 2 0 99 0 - #0-39-r #0-39-s;
#X floatatom 153 47 2 0 99 0 - #0-40-r #0-40-s;
45 #X floatatom 169 47 2 0 99 0 - #0-41-r #0-41-s;
#X floatatom 185 47 2 0 99 0 - #0-42-r #0-42-s;
#X floatatom 201 47 2 0 99 0 - #0-43-r #0-43-s;
#X floatatom 220 47 2 0 99 0 - #0-44-r #0-44-s;
#X floatatom 236 47 2 0 99 0 - #0-45-r #0-45-s;
50 #X floatatom 252 47 2 0 99 0 - #0-46-r #0-46-s;
#X floatatom 268 47 2 0 99 0 - #0-47-r #0-47-s;
#X floatatom 19 61 2 0 99 0 - #0-48-r #0-48-s;
#X floatatom 35 61 2 0 99 0 - #0-49-r #0-49-s;
#X floatatom 51 61 2 0 99 0 - #0-50-r #0-50-s;
55 #X floatatom 67 61 2 0 99 0 - #0-51-r #0-51-s;

```

```

#X floatatom 86 61 2 0 99 0 - #0-52-r #0-52-s;
#X floatatom 102 61 2 0 99 0 - #0-53-r #0-53-s;
#X floatatom 118 61 2 0 99 0 - #0-54-r #0-54-s;
#X floatatom 134 61 2 0 99 0 - #0-55-r #0-55-s;
60 #X floatatom 153 61 2 0 99 0 - #0-56-r #0-56-s;
#X floatatom 169 61 2 0 99 0 - #0-57-r #0-57-s;
#X floatatom 185 61 2 0 99 0 - #0-58-r #0-58-s;
#X floatatom 201 61 2 0 99 0 - #0-59-r #0-59-s;
#X floatatom 220 61 2 0 99 0 - #0-60-r #0-60-s;
65 #X floatatom 236 61 2 0 99 0 - #0-61-r #0-61-s;
#X floatatom 252 61 2 0 99 0 - #0-62-r #0-62-s;
#X floatatom 268 61 2 0 99 0 - #0-63-r #0-63-s;
#X floatatom 19 78 2 0 99 0 - #0-64-r #0-64-s;
#X floatatom 35 78 2 0 99 0 - #0-65-r #0-65-s;
70 #X floatatom 51 78 2 0 99 0 - #0-66-r #0-66-s;
#X floatatom 67 78 2 0 99 0 - #0-67-r #0-67-s;
#X floatatom 86 78 2 0 99 0 - #0-68-r #0-68-s;
#X floatatom 102 78 2 0 99 0 - #0-69-r #0-69-s;
#X floatatom 118 78 2 0 99 0 - #0-70-r #0-70-s;
75 #X floatatom 134 78 2 0 99 0 - #0-71-r #0-71-s;
#X floatatom 153 78 2 0 99 0 - #0-72-r #0-72-s;
#X floatatom 169 78 2 0 99 0 - #0-73-r #0-73-s;
#X floatatom 185 78 2 0 99 0 - #0-74-r #0-74-s;
#X floatatom 201 78 2 0 99 0 - #0-75-r #0-75-s;
80 #X floatatom 220 78 2 0 99 0 - #0-76-r #0-76-s;
#X floatatom 236 78 2 0 99 0 - #0-77-r #0-77-s;
#X floatatom 252 78 2 0 99 0 - #0-78-r #0-78-s;
#X floatatom 268 78 2 0 99 0 - #0-79-r #0-79-s;
#X floatatom 19 92 2 0 99 0 - #0-80-r #0-80-s;
85 #X floatatom 35 92 2 0 99 0 - #0-81-r #0-81-s;
#X floatatom 51 92 2 0 99 0 - #0-82-r #0-82-s;
#X floatatom 67 92 2 0 99 0 - #0-83-r #0-83-s;
#X floatatom 86 92 2 0 99 0 - #0-84-r #0-84-s;
#X floatatom 102 92 2 0 99 0 - #0-85-r #0-85-s;
90 #X floatatom 118 92 2 0 99 0 - #0-86-r #0-86-s;
#X floatatom 134 92 2 0 99 0 - #0-87-r #0-87-s;
#X floatatom 153 92 2 0 99 0 - #0-88-r #0-88-s;
#X floatatom 169 92 2 0 99 0 - #0-89-r #0-89-s;
#X floatatom 185 92 2 0 99 0 - #0-90-r #0-90-s;
95 #X floatatom 201 92 2 0 99 0 - #0-91-r #0-91-s;
#X floatatom 220 92 2 0 99 0 - #0-92-r #0-92-s;
#X floatatom 236 92 2 0 99 0 - #0-93-r #0-93-s;
#X floatatom 252 92 2 0 99 0 - #0-94-r #0-94-s;
#X floatatom 268 92 2 0 99 0 - #0-95-r #0-95-s;
100 #X floatatom 19 106 2 0 99 0 - #0-96-r #0-96-s;
#X floatatom 35 106 2 0 99 0 - #0-97-r #0-97-s;
#X floatatom 51 106 2 0 99 0 - #0-98-r #0-98-s;
#X floatatom 67 106 2 0 99 0 - #0-99-r #0-99-s;
#X floatatom 86 106 2 0 99 0 - #0-100-r #0-100-s;
105 #X floatatom 102 106 2 0 99 0 - #0-101-r #0-101-s;
#X floatatom 118 106 2 0 99 0 - #0-102-r #0-102-s;
#X floatatom 134 106 2 0 99 0 - #0-103-r #0-103-s;
#X floatatom 153 106 2 0 99 0 - #0-104-r #0-104-s;
#X floatatom 169 106 2 0 99 0 - #0-105-r #0-105-s;
110 #X floatatom 185 106 2 0 99 0 - #0-106-r #0-106-s;
#X floatatom 201 106 2 0 99 0 - #0-107-r #0-107-s;
#X floatatom 220 106 2 0 99 0 - #0-108-r #0-108-s;

```

```

#X floatatom 236 106 2 0 99 0 - #0-109-r #0-109-s;
#X floatatom 252 106 2 0 99 0 - #0-110-r #0-110-s;
115 #X floatatom 268 106 2 0 99 0 - #0-111-r #0-111-s;
#X floatatom 19 120 2 0 99 0 - #0-112-r #0-112-s;
#X floatatom 35 120 2 0 99 0 - #0-113-r #0-113-s;
#X floatatom 51 120 2 0 99 0 - #0-114-r #0-114-s;
#X floatatom 67 120 2 0 99 0 - #0-115-r #0-115-s;
120 #X floatatom 86 120 2 0 99 0 - #0-116-r #0-116-s;
#X floatatom 102 120 2 0 99 0 - #0-117-r #0-117-s;
#X floatatom 118 120 2 0 99 0 - #0-118-r #0-118-s;
#X floatatom 134 120 2 0 99 0 - #0-119-r #0-119-s;
#X floatatom 153 120 2 0 99 0 - #0-120-r #0-120-s;
125 #X floatatom 169 120 2 0 99 0 - #0-121-r #0-121-s;
#X floatatom 185 120 2 0 99 0 - #0-122-r #0-122-s;
#X floatatom 201 120 2 0 99 0 - #0-123-r #0-123-s;
#X floatatom 220 120 2 0 99 0 - #0-124-r #0-124-s;
#X floatatom 236 120 2 0 99 0 - #0-125-r #0-125-s;
130 #X floatatom 252 120 2 0 99 0 - #0-126-r #0-126-s;
#X floatatom 268 120 2 0 99 0 - #0-127-r #0-127-s;
#X floatatom 19 137 2 0 99 0 - #0-128-r #0-128-s;
#X floatatom 35 137 2 0 99 0 - #0-129-r #0-129-s;
#X floatatom 51 137 2 0 99 0 - #0-130-r #0-130-s;
135 #X floatatom 67 137 2 0 99 0 - #0-131-r #0-131-s;
#X floatatom 86 137 2 0 99 0 - #0-132-r #0-132-s;
#X floatatom 102 137 2 0 99 0 - #0-133-r #0-133-s;
#X floatatom 118 137 2 0 99 0 - #0-134-r #0-134-s;
#X floatatom 134 137 2 0 99 0 - #0-135-r #0-135-s;
140 #X floatatom 153 137 2 0 99 0 - #0-136-r #0-136-s;
#X floatatom 169 137 2 0 99 0 - #0-137-r #0-137-s;
#X floatatom 185 137 2 0 99 0 - #0-138-r #0-138-s;
#X floatatom 201 137 2 0 99 0 - #0-139-r #0-139-s;
#X floatatom 220 137 2 0 99 0 - #0-140-r #0-140-s;
145 #X floatatom 236 137 2 0 99 0 - #0-141-r #0-141-s;
#X floatatom 252 137 2 0 99 0 - #0-142-r #0-142-s;
#X floatatom 268 137 2 0 99 0 - #0-143-r #0-143-s;
#X floatatom 19 151 2 0 99 0 - #0-144-r #0-144-s;
#X floatatom 35 151 2 0 99 0 - #0-145-r #0-145-s;
150 #X floatatom 51 151 2 0 99 0 - #0-146-r #0-146-s;
#X floatatom 67 151 2 0 99 0 - #0-147-r #0-147-s;
#X floatatom 86 151 2 0 99 0 - #0-148-r #0-148-s;
#X floatatom 102 151 2 0 99 0 - #0-149-r #0-149-s;
#X floatatom 118 151 2 0 99 0 - #0-150-r #0-150-s;
155 #X floatatom 134 151 2 0 99 0 - #0-151-r #0-151-s;
#X floatatom 153 151 2 0 99 0 - #0-152-r #0-152-s;
#X floatatom 169 151 2 0 99 0 - #0-153-r #0-153-s;
#X floatatom 185 151 2 0 99 0 - #0-154-r #0-154-s;
#X floatatom 201 151 2 0 99 0 - #0-155-r #0-155-s;
160 #X floatatom 220 151 2 0 99 0 - #0-156-r #0-156-s;
#X floatatom 236 151 2 0 99 0 - #0-157-r #0-157-s;
#X floatatom 252 151 2 0 99 0 - #0-158-r #0-158-s;
#X floatatom 268 151 2 0 99 0 - #0-159-r #0-159-s;
#X floatatom 19 165 2 0 99 0 - #0-160-r #0-160-s;
165 #X floatatom 35 165 2 0 99 0 - #0-161-r #0-161-s;
#X floatatom 51 165 2 0 99 0 - #0-162-r #0-162-s;
#X floatatom 67 165 2 0 99 0 - #0-163-r #0-163-s;
#X floatatom 86 165 2 0 99 0 - #0-164-r #0-164-s;
#X floatatom 102 165 2 0 99 0 - #0-165-r #0-165-s;

```

```

170  #X floatatom 118 165 2 0 99 0 - #0-166-r #0-166-s;
      #X floatatom 134 165 2 0 99 0 - #0-167-r #0-167-s;
      #X floatatom 153 165 2 0 99 0 - #0-168-r #0-168-s;
      #X floatatom 169 165 2 0 99 0 - #0-169-r #0-169-s;
      #X floatatom 185 165 2 0 99 0 - #0-170-r #0-170-s;
175  #X floatatom 201 165 2 0 99 0 - #0-171-r #0-171-s;
      #X floatatom 220 165 2 0 99 0 - #0-172-r #0-172-s;
      #X floatatom 236 165 2 0 99 0 - #0-173-r #0-173-s;
      #X floatatom 252 165 2 0 99 0 - #0-174-r #0-174-s;
      #X floatatom 268 165 2 0 99 0 - #0-175-r #0-175-s;
180  #X floatatom 19 179 2 0 99 0 - #0-176-r #0-176-s;
      #X floatatom 35 179 2 0 99 0 - #0-177-r #0-177-s;
      #X floatatom 51 179 2 0 99 0 - #0-178-r #0-178-s;
      #X floatatom 67 179 2 0 99 0 - #0-179-r #0-179-s;
      #X floatatom 86 179 2 0 99 0 - #0-180-r #0-180-s;
185  #X floatatom 102 179 2 0 99 0 - #0-181-r #0-181-s;
      #X floatatom 118 179 2 0 99 0 - #0-182-r #0-182-s;
      #X floatatom 134 179 2 0 99 0 - #0-183-r #0-183-s;
      #X floatatom 153 179 2 0 99 0 - #0-184-r #0-184-s;
      #X floatatom 169 179 2 0 99 0 - #0-185-r #0-185-s;
190  #X floatatom 185 179 2 0 99 0 - #0-186-r #0-186-s;
      #X floatatom 201 179 2 0 99 0 - #0-187-r #0-187-s;
      #X floatatom 220 179 2 0 99 0 - #0-188-r #0-188-s;
      #X floatatom 236 179 2 0 99 0 - #0-189-r #0-189-s;
      #X floatatom 252 179 2 0 99 0 - #0-190-r #0-190-s;
195  #X floatatom 268 179 2 0 99 0 - #0-191-r #0-191-s;
      #X floatatom 19 196 2 0 99 0 - #0-192-r #0-192-s;
      #X floatatom 35 196 2 0 99 0 - #0-193-r #0-193-s;
      #X floatatom 51 196 2 0 99 0 - #0-194-r #0-194-s;
      #X floatatom 67 196 2 0 99 0 - #0-195-r #0-195-s;
200  #X floatatom 86 196 2 0 99 0 - #0-196-r #0-196-s;
      #X floatatom 102 196 2 0 99 0 - #0-197-r #0-197-s;
      #X floatatom 118 196 2 0 99 0 - #0-198-r #0-198-s;
      #X floatatom 134 196 2 0 99 0 - #0-199-r #0-199-s;
      #X floatatom 153 196 2 0 99 0 - #0-200-r #0-200-s;
205  #X floatatom 169 196 2 0 99 0 - #0-201-r #0-201-s;
      #X floatatom 185 196 2 0 99 0 - #0-202-r #0-202-s;
      #X floatatom 201 196 2 0 99 0 - #0-203-r #0-203-s;
      #X floatatom 220 196 2 0 99 0 - #0-204-r #0-204-s;
      #X floatatom 236 196 2 0 99 0 - #0-205-r #0-205-s;
210  #X floatatom 252 196 2 0 99 0 - #0-206-r #0-206-s;
      #X floatatom 268 196 2 0 99 0 - #0-207-r #0-207-s;
      #X floatatom 19 210 2 0 99 0 - #0-208-r #0-208-s;
      #X floatatom 35 210 2 0 99 0 - #0-209-r #0-209-s;
      #X floatatom 51 210 2 0 99 0 - #0-210-r #0-210-s;
215  #X floatatom 67 210 2 0 99 0 - #0-211-r #0-211-s;
      #X floatatom 86 210 2 0 99 0 - #0-212-r #0-212-s;
      #X floatatom 102 210 2 0 99 0 - #0-213-r #0-213-s;
      #X floatatom 118 210 2 0 99 0 - #0-214-r #0-214-s;
      #X floatatom 134 210 2 0 99 0 - #0-215-r #0-215-s;
220  #X floatatom 153 210 2 0 99 0 - #0-216-r #0-216-s;
      #X floatatom 169 210 2 0 99 0 - #0-217-r #0-217-s;
      #X floatatom 185 210 2 0 99 0 - #0-218-r #0-218-s;
      #X floatatom 201 210 2 0 99 0 - #0-219-r #0-219-s;
      #X floatatom 220 210 2 0 99 0 - #0-220-r #0-220-s;
225  #X floatatom 236 210 2 0 99 0 - #0-221-r #0-221-s;
      #X floatatom 252 210 2 0 99 0 - #0-222-r #0-222-s;

```

```

#X floatatom 268 210 2 0 99 0 - #0-223-r #0-223-s;
#X floatatom 19 224 2 0 99 0 - #0-224-r #0-224-s;
#X floatatom 35 224 2 0 99 0 - #0-225-r #0-225-s;
230 #X floatatom 51 224 2 0 99 0 - #0-226-r #0-226-s;
#X floatatom 67 224 2 0 99 0 - #0-227-r #0-227-s;
#X floatatom 86 224 2 0 99 0 - #0-228-r #0-228-s;
#X floatatom 102 224 2 0 99 0 - #0-229-r #0-229-s;
#X floatatom 118 224 2 0 99 0 - #0-230-r #0-230-s;
235 #X floatatom 134 224 2 0 99 0 - #0-231-r #0-231-s;
#X floatatom 153 224 2 0 99 0 - #0-232-r #0-232-s;
#X floatatom 169 224 2 0 99 0 - #0-233-r #0-233-s;
#X floatatom 185 224 2 0 99 0 - #0-234-r #0-234-s;
#X floatatom 201 224 2 0 99 0 - #0-235-r #0-235-s;
240 #X floatatom 220 224 2 0 99 0 - #0-236-r #0-236-s;
#X floatatom 236 224 2 0 99 0 - #0-237-r #0-237-s;
#X floatatom 252 224 2 0 99 0 - #0-238-r #0-238-s;
#X floatatom 268 224 2 0 99 0 - #0-239-r #0-239-s;
#X floatatom 19 238 2 0 99 0 - #0-240-r #0-240-s;
245 #X floatatom 35 238 2 0 99 0 - #0-241-r #0-241-s;
#X floatatom 51 238 2 0 99 0 - #0-242-r #0-242-s;
#X floatatom 67 238 2 0 99 0 - #0-243-r #0-243-s;
#X floatatom 86 238 2 0 99 0 - #0-244-r #0-244-s;
#X floatatom 102 238 2 0 99 0 - #0-245-r #0-245-s;
250 #X floatatom 118 238 2 0 99 0 - #0-246-r #0-246-s;
#X floatatom 134 238 2 0 99 0 - #0-247-r #0-247-s;
#X floatatom 153 238 2 0 99 0 - #0-248-r #0-248-s;
#X floatatom 169 238 2 0 99 0 - #0-249-r #0-249-s;
#X floatatom 185 238 2 0 99 0 - #0-250-r #0-250-s;
255 #X floatatom 201 238 2 0 99 0 - #0-251-r #0-251-s;
#X floatatom 220 238 2 0 99 0 - #0-252-r #0-252-s;
#X floatatom 236 238 2 0 99 0 - #0-253-r #0-253-s;
#X floatatom 252 238 2 0 99 0 - #0-254-r #0-254-s;
#X floatatom 268 238 2 0 99 0 - #0-255-r #0-255-s;
260 #N canvas 0 0 450 300 \ $0-guts 0;
#X obj 16 16 inlet;
#X obj 17 203 outlet;
#X obj 166 15 inlet;
#X obj 16 45 unpack f f;
265 #X obj 16 73 makefilename \ $0-%d-r;
#X obj 16 103 pack s f;
#X obj 166 45 unpack f f;
#X obj 166 73 makefilename \ $0-%d-r;
#X obj 166 103 pack s f;
270 #X msg 16 128 \; \ $1 \ $2;
#X msg 166 126 \; \ $1 set \ $2;
#X obj 17 174 nreceive \ $0-%d-s 0 256;
#X connect 0 0 3 0;
#X connect 2 0 6 0;
275 #X connect 3 0 4 0;
#X connect 3 1 5 1;
#X connect 4 0 5 0;
#X connect 5 0 9 0;
#X connect 6 0 7 0;
280 #X connect 6 1 8 1;
#X connect 7 0 8 0;
#X connect 8 0 10 0;
#X connect 11 0 1 0;

```

```

#X restore 327 57 pd \$0-guts;
285 #X obj 327 35 inlet;
#X obj 392 35 inlet;
#X obj 327 79 outlet;
#X connect 257 0 260 0;
#X connect 258 0 257 0;
290 #X connect 259 0 257 1;
#X coords 0 -1 1 1 282 252 1 10 10;

```

25 snare~.pd

```

#N canvas 0 0 834 579 10;
#X obj 19 17 inlet;
#X obj 19 42 select 0;
#X obj 51 119 noise~;
5 #X obj 186 166 vline~;
#X obj 185 190 *~;
#X obj 185 210 *~;
#X obj 55 258 *~;
#X obj 104 160 vline~;
10 #X obj 54 206 vcf~ 2.5;
#X obj 110 179 vline~;
#X obj 60 225 vcf~ 2.5;
#X obj 61 139 noise~;
#X obj 86 259 *~;
15 #X obj 216 186 vline~;
#X obj 215 210 *~;
#X obj 215 230 *~;
#X obj 124 49 t f f f f;
#X obj 95 93 * 1000;
20 #X msg 219 117 1 1 \, 0 \$1 1;
#X msg 240 140 1 1 \, 0 \$1 1;
#X obj 236 88 * 500;
#X obj 196 88 * 600;
#X obj 145 93 * 800;
25 #X msg 104 119 1500 1 \, 300 \$1 1;
#X msg 110 138 2500 1 \, 100 \$1 1;
#X obj 64 280 +~;
#X obj 489 120 hilbert~;
#X obj 483 158 complex-mod~;
30 #X obj 561 136 sig~ 175;
#X obj 484 41 phasor~ 111;
#X obj 574 157 complex-mod~;
#X obj 652 134 sig~ 224;
#X obj 330 159 osc~ 330;
35 #X obj 421 159 osc~ 180;
#X obj 605 271 *~;
#X obj 490 270 *~;
#X obj 425 269 *~;
#X obj 349 205 vline~;
40 #X obj 333 267 *~;
#X obj 348 227 *~;
#X obj 349 247 *~;
#X obj 442 205 vline~;
#X obj 441 227 *~;
45 #X obj 442 247 *~;
#X obj 529 205 vline~;

```

```

#X obj 528 227 *~;
#X obj 529 247 *~;
#X obj 612 205 vline~;
50 #X obj 611 227 *~;
#X obj 612 247 *~;
#X msg 349 184 1 1 \, 0 \$1 1;
#X msg 442 184 1 1 \, 0 \$1 1;
#X msg 529 184 1 1 \, 0 \$1 1;
55 #X msg 612 184 1 1 \, 0 \$1 1;
#X msg 538 92 bang;
#X obj 321 87 * 500;
#X obj 361 89 * 800;
#X obj 401 87 * 1000;
60 #X obj 418 65 * 1600;
#X floatatom 65 17 5 0 0 0 - - -;
#X obj 263 498 outlet~;
#X obj 427 497 outlet~;
#X obj 484 67 expr~ if($v1<0.5 \, $v1-0.5 \, 0.5-$v1);
65 #X obj 731 129 noise~;
#X obj 866 176 vline~;
#X obj 865 200 *~;
#X obj 865 220 *~;
#X obj 735 268 *~;
70 #X obj 784 170 vline~;
#X obj 734 216 vcf~ 2.5;
#X obj 790 189 vline~;
#X obj 740 235 vcf~ 2.5;
#X obj 741 149 noise~;
75 #X obj 766 269 *~;
#X obj 896 196 vline~;
#X obj 895 220 *~;
#X obj 895 240 *~;
#X obj 804 59 t f f f f;
80 #X obj 775 103 * 1000;
#X msg 899 127 1 1 \, 0 \$1 1;
#X msg 920 150 1 1 \, 0 \$1 1;
#X obj 916 98 * 500;
#X obj 876 98 * 600;
85 #X obj 825 103 * 800;
#X msg 784 129 1500 1 \, 300 \$1 1;
#X msg 790 148 2500 1 \, 100 \$1 1;
#X obj 744 290 +~;
#X obj 323 42 t f f f f b f;
90 #X obj 401 140 + 180;
#X obj 342 139 + 330;
#X obj 124 18 / 200;
#X obj 379 110 * 100;
#X obj 448 347 *~ 0.5;
95 #X obj 263 424 expr~ tanh($v2*5+4*$v1) \; tanh($v2*5+4*$v3);
#X connect 0 0 1 0;
#X connect 1 1 90 0;
#X connect 2 0 8 0;
#X connect 3 0 4 0;
100 #X connect 3 0 4 1;
#X connect 4 0 5 0;
#X connect 4 0 5 1;
#X connect 5 0 6 1;

```

```

#X connect 6 0 25 0;
105 #X connect 7 0 8 1;
#X connect 8 0 6 0;
#X connect 9 0 10 1;
#X connect 10 0 12 0;
#X connect 11 0 10 0;
110 #X connect 12 0 25 1;
#X connect 13 0 14 0;
#X connect 13 0 14 1;
#X connect 14 0 15 0;
#X connect 14 0 15 1;
115 #X connect 15 0 12 1;
#X connect 16 0 17 0;
#X connect 16 1 22 0;
#X connect 16 2 21 0;
#X connect 16 3 20 0;
120 #X connect 17 0 23 0;
#X connect 18 0 3 0;
#X connect 19 0 13 0;
#X connect 20 0 19 0;
#X connect 21 0 18 0;
125 #X connect 22 0 24 0;
#X connect 23 0 7 0;
#X connect 24 0 9 0;
#X connect 25 0 93 0;
#X connect 26 0 27 0;
130 #X connect 26 0 30 0;
#X connect 26 1 27 1;
#X connect 26 1 30 1;
#X connect 27 0 35 0;
#X connect 28 0 27 2;
135 #X connect 29 0 62 0;
#X connect 30 0 34 0;
#X connect 31 0 30 2;
#X connect 32 0 38 0;
#X connect 33 0 36 0;
140 #X connect 34 0 92 0;
#X connect 35 0 92 0;
#X connect 36 0 92 0;
#X connect 37 0 39 0;
#X connect 37 0 39 1;
145 #X connect 38 0 92 0;
#X connect 39 0 40 0;
#X connect 39 0 40 1;
#X connect 40 0 38 1;
#X connect 41 0 42 0;
150 #X connect 41 0 42 1;
#X connect 42 0 43 0;
#X connect 42 0 43 1;
#X connect 43 0 36 1;
#X connect 44 0 45 0;
155 #X connect 44 0 45 1;
#X connect 45 0 46 0;
#X connect 45 0 46 1;
#X connect 46 0 35 1;
#X connect 47 0 48 0;
160 #X connect 47 0 48 1;
```

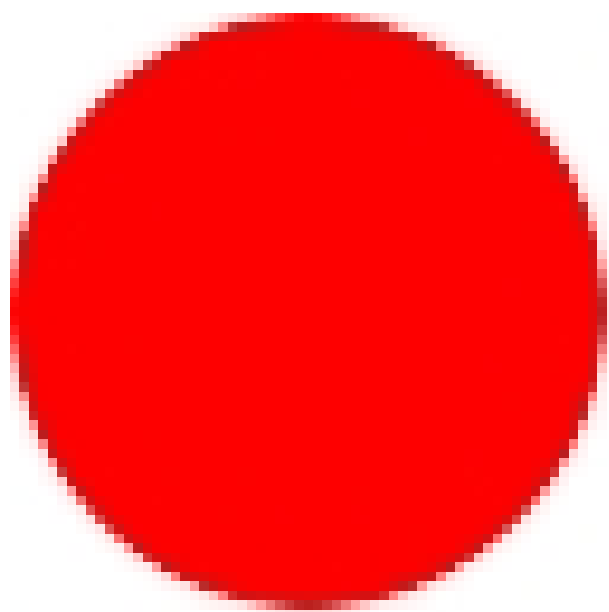
```
#X connect 48 0 49 0;
#X connect 48 0 49 1;
#X connect 49 0 34 1;
#X connect 50 0 37 0;
165 #X connect 51 0 41 0;
#X connect 52 0 44 0;
#X connect 53 0 47 0;
#X connect 54 0 26 1;
#X connect 55 0 50 0;
170 #X connect 56 0 51 0;
#X connect 57 0 52 0;
#X connect 58 0 53 0;
#X connect 59 0 1 0;
#X connect 62 0 26 0;
175 #X connect 63 0 69 0;
#X connect 64 0 65 0;
#X connect 64 0 65 1;
#X connect 65 0 66 0;
#X connect 65 0 66 1;
180 #X connect 66 0 67 1;
#X connect 67 0 86 0;
#X connect 68 0 69 1;
#X connect 69 0 67 0;
#X connect 70 0 71 1;
185 #X connect 71 0 73 0;
#X connect 72 0 71 0;
#X connect 73 0 86 1;
#X connect 74 0 75 0;
#X connect 74 0 75 1;
190 #X connect 75 0 76 0;
#X connect 75 0 76 1;
#X connect 76 0 73 1;
#X connect 77 0 78 0;
#X connect 77 1 83 0;
195 #X connect 77 2 82 0;
#X connect 77 3 81 0;
#X connect 78 0 84 0;
#X connect 79 0 64 0;
#X connect 80 0 74 0;
200 #X connect 81 0 80 0;
#X connect 82 0 79 0;
#X connect 83 0 85 0;
#X connect 84 0 68 0;
#X connect 85 0 70 0;
205 #X connect 86 0 93 2;
#X connect 87 0 55 0;
#X connect 87 1 56 0;
#X connect 87 2 57 0;
#X connect 87 3 58 0;
210 #X connect 87 4 54 0;
#X connect 87 5 91 0;
#X connect 88 0 33 0;
#X connect 89 0 32 0;
#X connect 90 0 16 0;
215 #X connect 90 0 77 0;
#X connect 90 0 87 0;
#X connect 91 0 89 0;
```

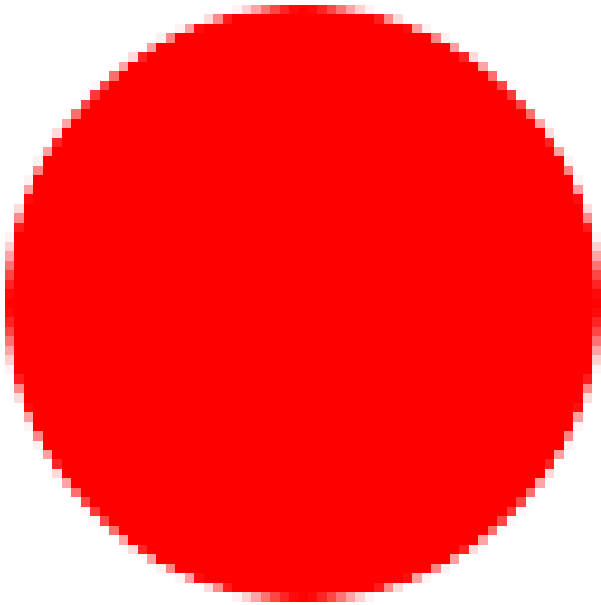
```
220 #X connect 91 0 88 0;  
#X connect 92 0 93 1;  
#X connect 93 0 60 0;  
#X connect 93 1 61 0;
```

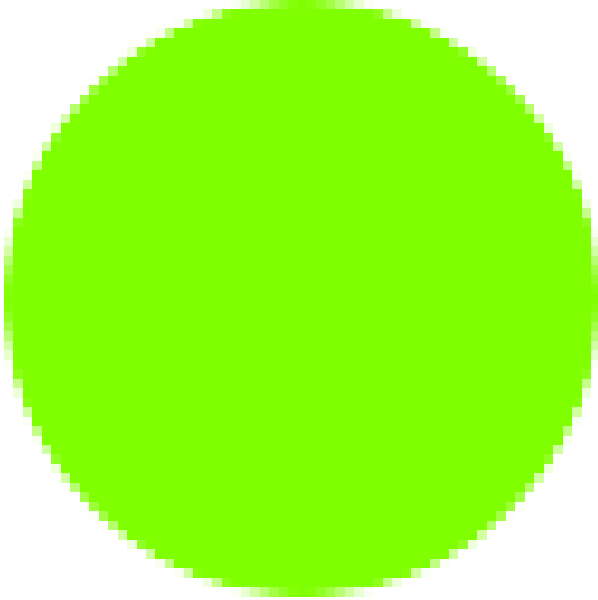
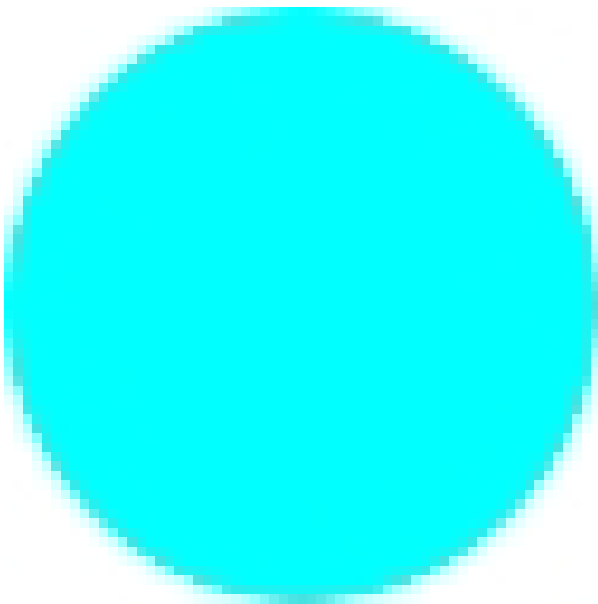
26 start.sh

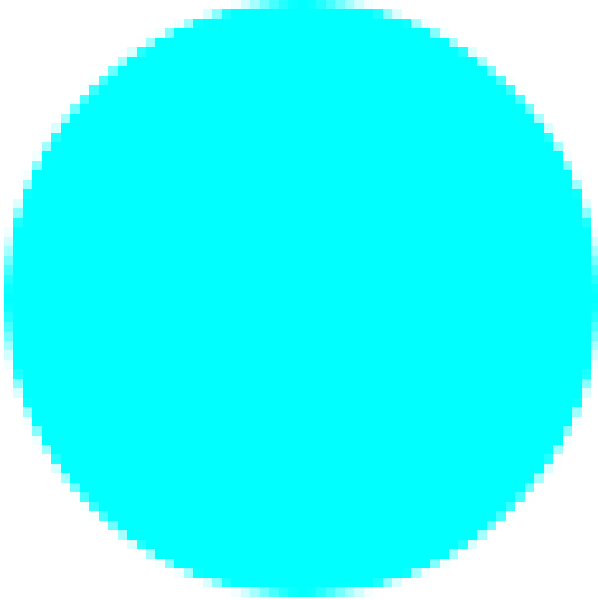
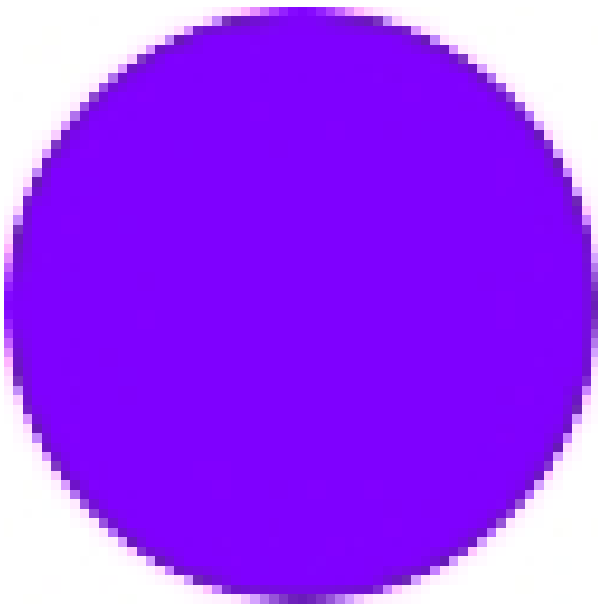
```
#!/bin/bash  
pd -stderr -verbose \  
-path ../analogue/ \  
-path ../hsex/ -lib hsex -path . \  
5 -open balls.pd @$@ >> /mnt/hd1/3/claude/SHUNT/performance2.log 2>&1
```

27 textures/0.jpg



28 textures/0.png**29** textures/1.jpg

30 textures/1.png**31 textures/2.jpg**

32 textures/2.png**33 textures/3.jpg**

34 textures/3.png

