

```

1  #include<windows.h>
2  #include<GL/glut.h>
3  #include <time.h>
4  double w=1280,h=720;
5  double view[3]={2,2,12.9};
6  double look[3]={2,2,2};
7  int flag=-1;
8  void steps(void);
9  void window(void);
10 void sgate(void);
11 void gate(void);
12 double angle=0,speed=5,maino=0,romo=0,tro=0,mgo=0,sgo=0;
13 //declarating quadric objects
14 GLUQuadricObj *Cylinder;
15 GLUQuadricObj *Disk;
16 struct tm *newtime;
17 time_t ltime;
18 GLfloat angle1;
19 //initialisation
20 void myinit(void)
21 {
22  glMatrixMode(GL_PROJECTION);
23  glLoadIdentity();
24  glFrustum(-1.0,1.0,-1*w/h,1*w/h,1,200.0);
25  glMatrixMode(GL_MODELVIEW);
26  glLoadIdentity();
27  //defining new quadric object
28  Cylinder = gluNewQuadric();
29  //to set drawing style
30  gluQuadricDrawStyle( Cylinder, GLU_FILL);
31  //to set automatic normals
32  gluQuadricNormals( Cylinder, GLU_SMOOTH);
33  Disk = gluNewQuadric();
34  gluQuadricDrawStyle( Disk, GLU_FILL);
35  gluQuadricNormals( Disk, GLU_SMOOTH);
36  GLfloat gam[]={0.2,.2,.2,1};
37  glLightModelfv(GL_LIGHT_MODEL_AMBIENT,gam);
38  }
39  //set material property
40  void matprop(GLfloat amb[],GLfloat dif[],GLfloat spec[],GLfloat shi[])
41  {
42  glMaterialfv(GL_FRONT_AND_BACK, GL_AMBIENT, amb);
43  glMaterialfv(GL_FRONT_AND_BACK, GL_DIFFUSE, dif);
44  glMaterialfv(GL_FRONT_AND_BACK, GL_SPECULAR, spec);
45  glMaterialfv(GL_FRONT_AND_BACK, GL_SHININESS, shi);
46  }
47  //to create wall
48  void wall(double thickness)
49  {
50  glPushMatrix();
51  glTranslated(2,.5*thickness,2);
52  glScaled(4.0,thickness,4.0);
53  glutSolidCube(1.0);
54  glPopMatrix();
55  }
56  //to create compound wall
57  void wall2(double thickness)
58  {
59  glPushMatrix();
60  glTranslated(.8,.5*thickness*4,3.5);
61  glScaled(1.6,thickness*4,7.0);
62  glutSolidCube(1.0);
63  glPopMatrix();
64  }
65  //to create earth
66  void earth(void)

```

```

67 {
68 GLfloat ambient[]={1,0,0,1};
69 GLfloat specular[]={0,1,1,1};
70 GLfloat diffuse[]={.5,.5,.5,1};
71 GLfloat shininess[]={50};
72 matprop(ambient,diffuse,specular,shininess);
73 GLfloat lightIntensity[]={.7,.7,.7,1};
74 GLfloat light_position[]={2,5,-3,0};
75 glLightfv(GL_LIGHT0,GL_POSITION,light_position);
76 glLightfv(GL_LIGHT0,GL_DIFFUSE,lightIntensity);
77 glPushMatrix();
78 glTranslated(0,-.25,0);
79 glScaled(10000,.5,1000000);
80 glutSolidCube(1.0);
81 glPopMatrix();
82 glFlush();
83 }
84 void compound(void)
85 {
86 GLfloat ambient[]={1,0,0,1};
87 GLfloat specular[]={0,1,1,1};
88 GLfloat diffuse[]={.7,1,.7,1};
89 GLfloat shininess[]={50};
90 matprop(ambient,diffuse,specular,shininess);
91 GLfloat lightIntensity[]={.7,.7,.7,1};
92 GLfloat light_position[]={2,6,1.5,0};
93 glLightfv(GL_LIGHT0,GL_POSITION,light_position);
94 glLightfv(GL_LIGHT0,GL_DIFFUSE,lightIntensity);
95 //left wall of compound
96 glPushMatrix();
97 glPushMatrix();
98 glTranslated(-4,0,-1-.04);
99 glRotated(90.0,0,0,1);
100 wall2(0.08);
101 glPopMatrix();
102 //right wall of compound
103 glPushMatrix();
104 glTranslated(8,0,-1-.02);
105 glRotated(90.0,0,0,1);
106 wall2(0.08);
107 glPopMatrix();
108 //back wall of compound
109 glPushMatrix();
110 glTranslated(2,.8,-1);
111 glRotated(-90,1,0,0);
112 glScaled(12,.02*4,1.6);
113 glutSolidCube(1.0);
114 glPopMatrix();
115 //front left wall of compound
116 glPushMatrix();
117 glTranslated(-3,.8,6-.08);
118 glRotated(-90,1,0,0);
119 glScaled(2,.02*4,1.6);
120 glutSolidCube(1.0);
121 glPopMatrix();
122 //front middle wall of compound
123 glPushMatrix();
124 glTranslated(2.5,.8,6-.08);
125 glRotated(-90,1,0,0);
126 glScaled(6,.02*4,1.6);
127 glutSolidCube(1.0);
128 glPopMatrix();
129 //front right wall of compound
130 glPushMatrix();
131 glTranslated(7,.8,6-.08);
132 glRotated(-90,1,0,0);

```

```

133  glScaled(2,.02*4,1.6);
134  glutSolidCube(1.0);
135  glPopMatrix();
136  glPopMatrix();
137  GLfloat ambient2[]={0,1,0,1};
138  GLfloat specular2[]={1,1,1,1};
139  GLfloat diffuse2[]={.2,.6,0.1,1};
140  GLfloat shininess2[]={50};
141  matprop(ambient2,diffuse2,specular2,shininess2);
142  //floor
143  glPushMatrix();
144  glTranslated(-4,-0.05,-1);
145  glScaled(3,3,1.7);
146  wall(0.08);
147  glPopMatrix();
148  gate();
149  sgate();
150  glFlush();
151  }
152  void room()
153  {
154  GLfloat ambient1[]={1,0,1,1};
155  GLfloat specular1[]={1,1,1,1};
156  GLfloat diffuse1[]={0.5,0.5,0.5,1};
157  GLfloat mat_shininess[]={50};
158  matprop(ambient1,diffuse1,specular1,mat_shininess);
159  glPushMatrix();
160  glTranslated(.5,4,.5);
161  //roof
162  glPushMatrix();
163  glTranslated(-.02*4,.7*3.9,-.02*4);
164  glScaled(.6+.02,1.5,.5+.02+.1);
165  wall(0.08);
166  glPopMatrix();
167  GLfloat ambient2[]={1,0,0,1};
168  GLfloat specular2[]={1,1,1,1};
169  GLfloat diffuse2[]={1,1,.7,1};
170  GLfloat shininess1[]={50};
171  matprop(ambient2,diffuse2,specular2,shininess1);
172  //left wall
173  glPushMatrix();
174  glTranslated(0,0,-.02);
175  glScaled(1,.7,.5);
176  glRotated(90.0,0,0,1);
177  wall(0.08);
178  glPopMatrix();
179  //right wall
180  glPushMatrix();
181  glTranslated(2.4,0,-.02);
182  glScaled(1,.7,.5);
183  glRotated(90.0,0,0,1);
184  wall(0.08);
185  glPopMatrix();
186  //back wall
187  glPushMatrix();
188  glTranslated(-.08,0,0);
189  glScaled(.62,.7,1);
190  glRotated(-90.0,1,0,0);
191  wall(0.08);
192  glPopMatrix();
193  //front wall
194  glPushMatrix();
195  glTranslated(-0.08,0,2);
196  glScaled(.5,.7,1);
197  glRotated(-90.0,1,0,0);
198  wall(0.08);

```

```

199 glPopMatrix();
200 //wall above the room door
201 glPushMatrix();
202 glTranslated(1.9,.7*3,2);
203 glScaled(.11,.7*.25,1);
204 glRotated(-90.0,1,0,0);
205 wall(0.08);
206 glPopMatrix();
207 GLfloat ambient[]={1,0.5,.5,1};
208 GLfloat specular[]={1,1,1,1};
209 GLfloat diffuse[]={1,0.5,0.5,1};
210 matprop(ambient,diffuse,specular,mat_shininess);
211 //door
212 glPushMatrix();
213 glTranslated(2.3,0,(2-.05));
214 glRotated(-tro,0,1,0);
215 glTranslated(-2.3,0,-(2-.05));
216 glPushMatrix();
217 glTranslated(1.927,0,2);
218 glScaled(.09,.525,1);
219 glRotated(-90.0,1,0,0);
220 wall(0.02);
221 glPopMatrix();
222 glPushMatrix();
223 glTranslated(2.3,0,2-.05);
224 glScaled(.6,.7,.8);
225 glRotated(-90,1,0,0);
226 gluCylinder(Cylinder, 0.05, 0.05, 3, 16, 16);
227 glPopMatrix();
228 glPopMatrix();
229 glPopMatrix();
230 }
231 void tankwall(float thk)
232 {
233 glTranslated(.5,.5*thk,.5);
234 glScaled(1,thk,1);
235 glutSolidCube(1);
236 }
237 void watertank(void)
238 {
239 float thk=.04,hght=1,width=1,bdth=1;
240 GLfloat ambient1[]={.5,0,1,1};
241 GLfloat specular1[]={1,1,1,1};
242 GLfloat diffuse1[]={.5,.8,1,1};
243 GLfloat mat_shininess[]={50};
244 matprop(ambient1,diffuse1,specular1,mat_shininess);
245 glPushMatrix();
246 glTranslated(1.5,4+4*.7,1.5);
247 glScaled(.8,.8,.8);
248 //tank floor
249 glPushMatrix();
250 glScaled(width,1,bdth);
251 tankwall(thk);
252 glPopMatrix();
253 //tank left wall
254 glPushMatrix();
255 glScaled(1,hght,bdth);
256 glRotated(90.0,0,0,1);
257 tankwall(thk);
258 glPopMatrix();
259 //tank right wall
260 glPushMatrix();
261 glTranslated(width,0,0);
262 glScaled(1,hght,bdth);
263 glRotated(90.0,0,0,1);
264 tankwall(thk);

```

```

265 glPopMatrix();
266 //tank back wall
267 glPushMatrix();
268 glScaled(wdth,hght,1);
269 glRotated(-90.0,1,0,0);
270 tankwall(0.04);
271 glPopMatrix();
272 //tank front wall
273 glPushMatrix();
274 glTranslated(0,0,bdth);
275 glScaled(wdth,hght,1);
276 glRotated(-90.0,1,0,0);
277 tankwall(0.04);
278 glPopMatrix();
279 //tank roof
280 glPushMatrix();
281 glTranslated(-thk,hght,0);
282 glScaled(wdth*.8,1,bdth);
283 tankwall(0.04);
284 glPopMatrix();
285 glPushMatrix();
286 glTranslated(wdth*.8-thk,hght,0);
287 glScaled(wdth*.2+thk,1,bdth*.6);
288 tankwall(0.04);
289 glPopMatrix();
290 glPopMatrix();
291 }
292 void terece(void)
293 {
294 GLfloat ambient1[]={1,0,1,1};
295 GLfloat specular1[]={1,1,1,1};
296 GLfloat diffusel[]={0.5,0.5,0.5,1};
297 GLfloat mat_shininess[]={50};
298 matprop(ambient1,diffusel,specular1,mat_shininess);
299 glPushMatrix();
300 glTranslated(0,4,0);
301 glScaled(1,.1,1);
302 //left wall
303 glPushMatrix();
304 glTranslated(0,0,-.02-.25);
305 glScaled(1,1,.95);
306 glRotated(90.0,0,0,1);
307 wall(0.08);
308 glPopMatrix();
309 //right wall
310 glPushMatrix();
311 glTranslated(6+.12,0,-.02-.27);
312 glScaled(1,1,1.1);
313 glRotated(90.0,0,0,1);
314 wall(0.08);
315 glPopMatrix();
316 //back wall
317 glPushMatrix();
318 glTranslated(-.08,0,-.21);
319 glScaled(1.5+.05,1,1);
320 glRotated(-90.0,1,0,0);
321 wall(0.08);
322 glPopMatrix();
323 //front wall
324 glPushMatrix();
325 glTranslated(-.08,0,4+.11);
326 glScaled(1.5+.05,1,1);
327 glRotated(-90.0,1,0,0);
328 wall(0.08);
329 glPopMatrix();
330 glPushMatrix();

```

```

331 glTranslated(-.04,2,4);
332 glScaled(.08,4,.1);
333 glutSolidCube(1);
334 glPopMatrix();
335 glPopMatrix();
336 }
337 void fanwing(float wingle)
338 {
339     glPushMatrix();
340     glRotated(90,1,0,0);
341     glRotated(90,0,1,0);
342     glScaled(1,15,1);
343     gluCylinder(Cylinder, .01, .01, 1, 4, 1);
344     glPopMatrix();
345 }
346 void fanbottom()
347 {
348     glPushMatrix();
349     glTranslated(1,3.2,1);
350     glPushMatrix();
351     glRotated(90,1,0,0);
352     gluCylinder(Cylinder, .2, .2, .05, 128, 16);
353     glPopMatrix();
354     glPushMatrix();
355     glTranslated(0,0.00025,0);
356     glRotated(90,1,0,0);
357     gluDisk(Disk,0,.2,32,16);
358     glPopMatrix();
359     glPushMatrix();
360     glTranslated(0,-.05,0);
361     glRotated(90,1,0,0);
362     gluCylinder(Cylinder, .2, .15, .1, 128, 16);
363     glPopMatrix();
364     glPushMatrix();
365     glTranslated(0,-.1,0);
366     glRotated(90,1,0,0);
367     gluDisk(Disk,0,.15,32,16);
368     glPopMatrix();
369     glPushMatrix();
370     glTranslated(0.1,0.0,0);
371     fanwing(.6);
372     glPopMatrix();
373     glPushMatrix();
374     glRotated(120,0,1,0);
375     glTranslated(.1,0,0);
376     fanwing(.6);
377     glPopMatrix();
378     glPushMatrix();
379     glRotated(240,0,1,0);
380     glTranslated(0.1,0.0,0);
381     fanwing(.6);
382     glPopMatrix();
383     glPopMatrix();
384 }
385 void fan(void)
386 {
387     glPushMatrix();
388     glTranslated(2.5,1.9,0);
389     glScaled(.5,.5,.5);
390     GLfloat mat_ambient[]={.5,0,0,1};
391     GLfloat mat_specular[]={0,1,1,0};
392     GLfloat mat_diffuse[]={.8,1,.8,1};
393     GLfloat mat_shininess[]={50};
394     glMaterialfv(GL_FRONT, GL_AMBIENT, mat_ambient);
395     glMaterialfv(GL_FRONT, GL_DIFFUSE, mat_diffuse);
396     glMaterialfv(GL_FRONT, GL_SPECULAR, mat_specular);

```

```

397 glMaterialfv(GL_FRONT, GL_SHININESS, mat_shininess);
398 if(flag==1)
399 {
400 glPushMatrix();
401 fanbottom();
402 glPopMatrix();
403 }
404 else
405 {
406 angle+=speed;
407 glPushMatrix();
408 glTranslated(1,0,1);
409 glRotated(angle,0,1,0);
410 glTranslated(-1,0,-1);
411 fanbottom();
412 glPopMatrix();
413 }
414 glPushMatrix();
415 glTranslatef(1,3.3,1);
416 glRotated(-90,1,0,0);
417 gluCylinder(Cylinder, .1, 0.005, .25, 16, 16);
418 glPopMatrix();
419 glPushMatrix();
420 glTranslatef(1,4,1);
421 glRotated(90,1,0,0);
422 gluCylinder(Cylinder, .006, 0.006, .6, 16, 16);
423 glPopMatrix();
424 glPushMatrix();
425 glTranslatef(1,3.96,1);
426 glRotated(90,1,0,0);
427 gluCylinder(Cylinder, .1, 0.005, .25, 16, 16);
428 glPopMatrix();
429 glPopMatrix();
430 if(flag==1)
431 glutPostRedisplay();
432 }
433 void tableg(float llen, float lthk)
434 {
435 glPushMatrix();
436 glRotated(-90,1,0,0);
437 gluCylinder(Cylinder, lthk, lthk, llen, 32, 32);
438 glPopMatrix();
439 }
440 void table(float tabwid, float tablen, float tabthk, float llen, float lthk)
441 {
442 glPushMatrix();
443 glPushMatrix();
444 glTranslated(0, llen, 0);
445 glScaled(tabwid, tabthk, tablen);
446 glutSolidCube(1);
447 glPopMatrix();
448 float dist1=.95*tablen/2-lthk/2;
449 float dist2=.95*tabwid/2-lthk/2;
450 // front right leg
451 glPushMatrix();
452 glTranslated(dist2, 0, dist1);
453 tableg(llen, lthk);
454 glPopMatrix();
455 //back right leg
456 glPushMatrix();
457 glTranslated(dist2, 0, -dist1);
458 tableg(llen, lthk);
459 glPopMatrix();
460 //back left leg
461 glPushMatrix();
462 glTranslated(-dist2, 0, -dist1);

```

```

463 tableg(llen,lthk);
464 glPopMatrix();
465 //front left leg
466 glPushMatrix();
467 glTranslated(-dist2,0,dist1);
468 tableg(llen,lthk);
469 glPopMatrix();
470 glPopMatrix();
471 }
472 void cot(float cwid,float clen,float cthk,float llen,float lthk)
473 {
474 GLfloat ambient1[]={1,0,.4,1};
475 GLfloat specular1[]={1,1,1,1};
476 GLfloat diffuse1[]={0.5,0.5,0.5,1};
477 GLfloat mat_shininess[]={50};
478 matprop(ambient1,diffuse1,specular1,mat_shininess);
479 glPushMatrix();
480 glTranslated(5.6,0,.5);
481 table(cwid,clen,cthk,llen,lthk);
482 glPushMatrix();
483 glTranslated(0,llen,clen/2);
484 GLdouble eqn[3] = {0.0,1.0, 0.0};
485 glPushMatrix();
486 glClipPlane(GL_CLIP_PLANE0, eqn);
487 //void glClipPlane(GLenum plane,const GLdouble *equation);
488 glEnable (GL_CLIP_PLANE0);//enable clip plane
489 gluDisk(Disk,0,cwid/2,32,32);
490 glPopMatrix();
491 glDisable(GL_CLIP_PLANE0);
492 glPopMatrix();
493 glPushMatrix();
494 glTranslated(0,llen,-clen/2);
495 glPushMatrix();
496 glClipPlane (GL_CLIP_PLANE0, eqn);
497 glEnable (GL_CLIP_PLANE0);
498 glScaled(1,1.5,1);
499 gluDisk(Disk,0,cwid/2,32,32);
500 glPopMatrix();
501 glDisable(GL_CLIP_PLANE0);
502 glPopMatrix();
503 glPopMatrix();
504 }
505 void cleg(float cllen,float clwid)
506 {
507 glRotated(90,1,0,0);
508 gluCylinder(Cylinder,clwid,clwid,cllen,32,32);
509 }
510 void chair(float cblen,float cbwid,float cbthk,float cllen,float clwid)
511 {
512 GLfloat ambient1[]={.5,1,.5,1};
513 GLfloat specular1[]={1,1,1,1};
514 GLfloat diffuse1[]={0.5,0.5,0.5,1};
515 GLfloat mat_shininess[]={50};
516 matprop(ambient1,diffuse1,specular1,mat_shininess);
517 glPushMatrix();
518 glTranslated(0,cllen,0);
519 //chair base
520 glPushMatrix();
521 glScaled(cblen,cbthk,cbwid);
522 glutSolidCube(1);
523 glPopMatrix();
524 float dist=cblen/2-clwid/2;
525 //chair legs
526 glPushMatrix();
527 glTranslated(dist,0,dist);
528 cleg(cllen,clwid);

```



```

529 glPopMatrix();
530 glPushMatrix();
531 glTranslated(-dist,0,dist);
532 cleg(cllen,clwid);
533 glPopMatrix();
534 glPushMatrix();
535 glTranslated(-dist,0,-dist);
536 cleg(cllen,clwid);
537 glPopMatrix();
538 glPushMatrix();
539 glTranslated(dist,0,-dist);
540 cleg(cllen,clwid);
541 glPopMatrix();
542 //base pipes
543 glPushMatrix();
544 glTranslated(-.085,-clwid/2,cbwid/3);
545 glRotated(90,0,1,0);
546 gluCylinder(Cylinder,-clwid,clwid,cblen,32,32);
547 glPopMatrix();
548 glPushMatrix();
549 glTranslated(-.085,clwid/2,-cbwid/3);
550 glRotated(90,0,1,0);
551 gluCylinder(Cylinder,clwid,clwid,cblen,32,32);
552 glPopMatrix();
553 //back support pipes
554 glPushMatrix();
555 glTranslated(-.085,-clwid/2,cbwid/3);
556 glRotated(-90,1,0,0);
557 gluCylinder(Cylinder,clwid,clwid,cllen,32,32);
558 glPopMatrix();
559 glPushMatrix();
560 glTranslated(-.085,-clwid/2,-cbwid/3);
561 glRotated(-90,1,0,0);
562 gluCylinder(Cylinder,clwid,clwid,cllen,32,32);
563 glPopMatrix();
564 //back support
565 glPushMatrix();
566 glTranslated(-cblen/2,cllen/2+cblen/2,0);
567 glRotated(90,0,0,1);
568 glScaled(cblen,.01,cbwid);
569 glutSolidCube(1);
570 glPopMatrix();
571 glPopMatrix();
572 }
573 void diningtable()
574 {
575 glPushMatrix();
576 glTranslated(3,0,1);
577 glScaled(1.5,1.5,1.5);
578 table(.3,.5,.025,.4,.005);
579 //front left chair
580 glPushMatrix();
581 glTranslated(-.1,0,.1);
582 chair(.15,.15,.02,.3,.005);
583 glPopMatrix();
584 //back left chair
585 glPushMatrix();
586 glTranslated(-.1,0,-.1);
587 chair(.15,.15,.02,.3,.005);
588 glPopMatrix();
589 //front right chair
590 glPushMatrix();
591 glTranslated(.1,0,.1);
592 glRotated(180,0,1,0);
593 chair(.15,.15,.02,.3,.005);
594 glPopMatrix();

```

```

595 //back right chair
596 glPushMatrix();
597 glTranslated(.1,0,-.1);
598 glRotated(180,0,1,0);
599 chair(.15,.15,.02,.3,.005);
600 glPopMatrix();
601 //back chair
602 glPushMatrix();
603 glTranslated(0,0,-.27);
604 glRotated(-90,0,1,0);
605 chair(.15,.15,.02,.3,.005);
606 glPopMatrix();
607 //front chair
608 glPushMatrix();
609 glTranslated(0,0,.27);
610 glRotated(90,0,1,0);
611 chair(.15,.15,.02,.3,.005);
612 glPopMatrix();
613 glPopMatrix();
614 }
615 void steps(void)
616 {
617     int i;
618     GLfloat ambient1[]={1,0,1,1};
619     GLfloat specular1[]={1,1,1,1};
620     GLfloat diffuse1[]={0.5,0.5,0.5,1};
621     GLfloat mat_shininess[]={50};
622     matprop(ambient1,diffuse1,specular1,mat_shininess);
623     glPushMatrix();
624     glTranslated(-.25,.1,.2);
625     for(i=0;i<19;i++)
626     {
627         glPushMatrix();
628         glTranslated(0,i*.2,i*.2);
629         glScaled(.4,.2,.3);
630         glutSolidCube(1);
631         glPopMatrix();
632     }
633     glPopMatrix();
634     glPushMatrix();
635     glRotated(-45,1,0,0);
636     glTranslated(-.45,.3,2.7);
637     glScaled(.04,1,5.4);
638     glutSolidCube(1);
639     glPopMatrix();
640     glPushMatrix();
641     glTranslated(-.45,4,3.6);
642     glScaled(.04,.8,.75);
643     glutSolidCube(1);
644     glPopMatrix();
645     glPushMatrix();
646     glTranslated(-.25,4,3.96);
647     glScaled(.4,.8,.04);
648     glutSolidCube(1);
649     glPopMatrix();
650 }
651 void sleg(float len,float thk)
652 {
653     glScaled(thk,len,thk);
654     glutSolidCube(1);
655 }
656 void solar(void)
657 {
658     GLfloat ambient1[]={.1,.1,.1,1};
659     GLfloat specular1[]={1,1,1,1};
660     GLfloat diffuse1[]={1,1,1,1};

```

```

661 GLfloat mat_shininess[]={50};
662 matprop(ambient1,diffuse1,specular1,mat_shininess);
663 GLfloat lightIntensity[]={.7,.7,.7,1};
664 GLfloat light_position[]={-20,4,60,0};
665 glLightfv(GL_LIGHT2,GL_POSITION,light_position);
666 glLightfv(GL_LIGHT2,GL_DIFFUSE,lightIntensity);
667 glEnable(GL_LIGHT2);
668 //base
669 glPushMatrix();
670 glTranslated(4,4,3);
671 glPushMatrix();
672 glTranslated(0.4,.4,0);
673 glScaled(1,.8,1);
674 glutSolidCube(1);
675 glPopMatrix();
676 GLfloat ambient2[]={.7,.7,.7,1};
677 GLfloat specular2[]={1,1,1,1};
678 GLfloat diffuse2[]={1,1,1,1};
679 matprop(ambient2,diffuse2,specular2,mat_shininess);
680 glPushMatrix();
681 glTranslated(0,.8,0);
682 glPushMatrix();
683 glTranslated(.6,.6,0);
684 gluCylinder(Cylinder,.1,.1,.4,32,32);
685 glPopMatrix();
686 GLfloat ambient3[]={1,0,.2,1};
687 GLfloat specular3[]={1,1,1,1};
688 GLfloat diffuse3[]={1,0,.5,1};
689 GLfloat mat_shininess3[]={50};
690 matprop(ambient3,diffuse3,specular3,mat_shininess3);
691 glPushMatrix();
692 glTranslated(.6,.6,0);
693 gluDisk(Disk,0,.1,32,32);
694 glPopMatrix();
695 glPushMatrix();
696 glTranslated(.6,.6,0.4);
697 gluDisk(Disk,0,.1,32,32);
698 glPopMatrix();
699 GLfloat ambient4[]={0,0,0,1};
700 GLfloat specular4[]={1,1,1,1};
701 GLfloat diffuse4[]={0,0,0,1};
702 GLfloat mat_shininess4[]={50};
703 matprop(ambient4,diffuse4,specular4,mat_shininess4);
704 glPushMatrix();
705 glTranslated(.5,.3,.05);
706 sleg(.6,.01);
707 glPopMatrix();
708 glPushMatrix();
709 glTranslated(.7,.3,.05);
710 sleg(.6,.01);
711 glPopMatrix();
712 glPushMatrix();
713 glTranslated(.5,.3,.35);
714 sleg(.6,.01);
715 glPopMatrix();
716 glPushMatrix();
717 glTranslated(.7,.3,.35);
718 sleg(.6,.01);
719 glPopMatrix();
720 glPushMatrix();
721 glRotated(45,0,0,1);
722 glTranslated(.3,.015,.2);
723 glScaled(.6,.03,.4);
724 glutSolidCube(1);
725 glPopMatrix();
726 glPushMatrix();

```

```

727 glTranslated(.4,.21,0);
728 sleg(.425,.01);
729 glPopMatrix();
730 glPushMatrix();
731 glTranslated(.4,.21,.4);
732 sleg(.425,.01);
733 glPopMatrix();
734 glPushMatrix();
735 glTranslated(.4,.4,0);
736 glRotated(30,0,0,1);
737 glRotated(90,0,1,0);
738 gluCylinder(Cylinder,.01,.01,.2,32,32);
739 glPopMatrix();
740 glPopMatrix();
741 glPopMatrix();
742 }
743
744 void myclock()
745 {
746 GLfloat mat_ambient[]={.4,.8,.4,1};
747 GLfloat mat_specular[]={1,1,1,1};
748 GLfloat mat_diffuse[]={0.4,.8,.4,1};
749 GLfloat mat_shininess[]={50};
750 matprop(mat_ambient,mat_diffuse,mat_specular,mat_shininess);
751 int hour_ticks , sec_ticks;
752 glPushMatrix();
753 glTranslated(2,3.2,-.02);
754 glScaled(.03,.06,.03);
755 glPushMatrix(); // Draw clock face
756
757 glTranslatef( 0, 0, 1.0);
758 gluDisk(Disk, 0, 7, 32, 16);
759 glPopMatrix();
760 GLfloat mat_ambien[]={1,0,0,1};
761 matprop(mat_ambien,mat_diffuse,mat_specular,mat_shininess);
762
763 glPushMatrix();
764 glTranslatef( 0, 0, 1.95);
765 gluDisk(Disk, 0, .8, 32, 16);
766 glPopMatrix();
767 GLfloat ambient[]={0,0,0,1};
768 GLfloat specular[]={1,1,1,1};
769 GLfloat diffuse[]={0,0,0,1};
770 matprop(ambient,diffuse,specular,mat_shininess);
771 // Draw hour hand
772 glPushMatrix();
773 glColor3f(1.0, 0.5, 0.5);
774 glTranslatef( 0, 0, 1.5);
775 glRotatef( -(360/12) * (newtime->tm_hour+newtime->tm_min/60.0), 0.0,0.0, 1.0);
776 glRotatef( -90, 1.0, 0.0, 0.0);
777 gluCylinder(Cylinder, 0.45, 0, 4, 16, 16);
778 glPopMatrix();
779 GLfloat ambient1[]={0,0,1,1};
780 GLfloat specular1[]={1,1,1,1};
781 GLfloat diffusel[]={0,0,1,1};
782 matprop(ambient1,diffusel,specular1,mat_shininess);
783 // Draw minute hand
784 glPushMatrix();
785 glColor3f(1.0, 0.5, 1.0);
786 glTranslatef( 0, 0, 1.25);
787 glRotatef( -(360/60) * newtime->tm_min, 0.0, 0.0, 1.0);
788 glRotatef(-90, 1.0, 0.0, 0.0);
789 gluCylinder(Cylinder, 0.4, 0, 6, 16, 16);
790 glPopMatrix();
791 GLfloat ambient2[]={1,0,0,1};
792 GLfloat specular2[]={1,1,1,1};

```

```

793 GLfloat diffuse2[]={1,0,0,1};
794 matprop(ambient2,diffuse2,specular2,mat_shininess);
795 // Draw second hand
796 glPushMatrix();
797 glTranslatef( 0, 0, 1);
798 glRotatef(-(360/60) * newtime->tm_sec, 0.0, 0.0, 1.0);
799 glRotatef( -90, 1.0, 0.0, 0.0);
800 gluCylinder(Cylinder, 0.3, 0, 6, 16, 16);
801 glPopMatrix();
802 GLfloat ambient3[]={1,1,1,1};
803 GLfloat specular3[]={1,1,1,1};
804 GLfloat diffuse3[]={1,0,1,1};
805
806 matprop(ambient3,diffuse3,specular3,mat_shininess);
807 for(hour_ticks = 0; hour_ticks < 12; hour_ticks++)
808 {
809 glPushMatrix();// Draw next arm axis.
810 glTranslatef(0.0, 0.0, 1);
811 glRotatef( (360/12) * hour_ticks, 0.0, 0.0, 1.0);
812 glTranslatef( 6.0, 0.0, 0.0);
813 glutSolidCube(.8);
814 glPopMatrix();
815 }
816 for(sec_ticks = 0; sec_ticks < 60; sec_ticks++)
817 {
818 glPushMatrix();
819 glTranslatef(0.0, 0.0, 1.1);
820 glRotatef( (360/60) * sec_ticks, 0.0, 0.0, 1.0);
821 glTranslatef(6.0, 0.0, 0.0);
822 glutSolidCube(0.25);
823 glPopMatrix();
824 }
825 glPopMatrix();
826 }
827 void window(void)
828 {
829 int i;
830 GLfloat lightIntensity[]={.7,.7,.7,1};
831 GLfloat light_position[]={-20,4,-60,0};
832 glLightfv(GL_LIGHT1,GL_POSITION,light_position);
833 glLightfv(GL_LIGHT1,GL_DIFFUSE,lightIntensity);
834 glEnable(GL_LIGHT1);
835 glPushMatrix();
836 glTranslated(3.185,1,3.95);
837 //left edge of window
838 glPushMatrix();
839 glTranslated(.02,1,.02);
840 glScaled(.04,2.2,.04);
841 glutSolidCube(1);
842 glPopMatrix();
843 //right edge
844 glPushMatrix();
845 glTranslated(1.6+.02,1,0.02);
846 glScaled(.04,2.2,.04);
847 glutSolidCube(1);
848 glPopMatrix();
849 //top edge
850 glPushMatrix();
851 glTranslated(.9,2+.02,0.02);
852 glScaled(1.8,.04,.04);
853 glutSolidCube(1);
854 glPopMatrix();
855 //bottom edge
856 glPushMatrix();
857 glTranslated(.8,.02,0.02);
858 glScaled(1.8,.04,.04);

```

```

859  glutSolidCube(1);
860  glPopMatrix();
861  for(i=1;i<=3;i++)
862  {
863      glPushMatrix();
864      glTranslated(.4*i,0,0);
865      glRotated(-90,1,0,0);
866      gluCylinder(Cylinder,.01,.01,2,32,32);
867      glPopMatrix();
868  }
869  for(i=1;i<=3;i++)
870  {
871      glPushMatrix();
872      glTranslated(.1+.4*i,0,0);
873      glRotated(-90,1,0,0);
874      gluCylinder(Cylinder,.01,.01,2,32,32);
875      glPopMatrix();
876  }
877  for(i=1;i<=4;i++)
878  {
879      glPushMatrix();
880      glTranslated(0,.4*i,0);
881      glRotated(90,0,1,0);
882      gluCylinder(Cylinder,.03,.03,1.6,32,32);
883      glPopMatrix();
884  }
885  glPopMatrix();
886  }
887  void gate(void)
888  {
889      int i;
890      GLfloat ambient1[]={1,.5,1,1};
891      GLfloat specular1[]={1,1,1,1};
892      GLfloat diffuse1[]={.5,.5,.5,1};
893      GLfloat mat_shininess[]={120};
894      matprop(ambient1,diffuse1,specular1,mat_shininess);
895      glPushMatrix();
896      //if flag mgo=1 the open the main gate
897      if(mgo==1)
898      glTranslated(1.5,0,0);
899      glTranslated(-1.3,0,6);
900      //top frame of the main gate
901      glPushMatrix();
902      glTranslated(0,1.5,0);
903      glScaled(1.7,.04,.04);
904      glutSolidCube(1);
905      glPopMatrix();
906      //bottom frame of main gate
907      glPushMatrix();
908      glTranslated(0,.05,0);
909      glScaled(1.7,.04,.04);
910      glutSolidCube(1);
911      glPopMatrix();
912      //left frame of the main gate
913      glPushMatrix();
914      glTranslated(-.8,.75,0);
915      glScaled(.04,1.5,.04);
916      glutSolidCube(1);
917      glPopMatrix();
918      //right frame of the main gate
919      glPushMatrix();
920      glTranslated(.8,.75,0);
921      glScaled(.04,1.5,.04);
922      glutSolidCube(1);
923      glPopMatrix();
924      //horizontal pipes of the main gate

```

```

925  for(i=1;i<=3;i++)
926  {
927      glPushMatrix();
928      glTranslated(-.85,.4*i,0);
929      glRotated(90,0,1,0);
930      gluCylinder(Cylinder,.02,.02,1.7,32,32);
931      glPopMatrix();
932  }
933  //vertical strips of the main gate
934  for(i=1;i<=5;i++)
935  {
936      glPushMatrix();
937      glTranslated(-.9+.3*i,.75,0);
938      glScaled(.2,1.5,.02);
939      glutSolidCube(1);
940      glPopMatrix();
941  }
942  glPopMatrix();
943  }
944  void sgate(void )
945  {
946      int i;
947      GLfloat ambient1[]={1,.5,1,1};
948      GLfloat specular1[]={1,1,1,1};
949      GLfloat diffuse1[]={.5,.5,.5,1};
950      GLfloat mat_shininess[]={120};
951      matprop(ambient1,diffuse1,specular1,mat_shininess);
952      glPushMatrix();
953      //to open the sub gate
954      glTranslated(5.75-.25,.05,6);
955      glRotated(sgo,0,1,0);
956      glTranslated(-5.75+.25,-.05,-6);
957      //to translate sub gate to required position
958      glTranslated(5.75,.05,6);
959      //top edge of the sub gate
960      glPushMatrix();
961      glTranslated(0,1.5,0);
962      glScaled(.5,.08,.08);
963      glutSolidCube(1);
964      glPopMatrix();
965      //bottom edge of the sub gate
966      glPushMatrix();
967      glTranslated(0,.05,0);
968      glScaled(.5,.08,.08);
969      glutSolidCube(1);
970      glPopMatrix();
971      //left edge of the sub gate
972      glPushMatrix();
973      glTranslated(-.25,.85,0);
974      glScaled(.04,1.7,.04);
975      glutSolidCube(1);
976      glPopMatrix();
977      //right edge of the sub gate
978      glPushMatrix();
979      glTranslated(.25,.8,0);
980      glScaled(.04,1.6,.04);
981      glutSolidCube(1);
982      glPopMatrix();
983      //vertical pipes of the sub gate
984      for(i=1;i<=4;i++)
985      {
986          glPushMatrix();
987          glTranslated(-.25+.1*i,0,0);
988          glRotated(-90,1,0,0);
989          gluCylinder(Cylinder,.01,.01,1.5,32,32);
990          glPopMatrix();

```

```

991 }
992 //horizontal pipes of the sub gate
993 for( i=1;i<=4;i++)
994 {
995     glPushMatrix();
996     glTranslated(-.25,.05+.3*i,0);
997     glRotated(90,0,1,0);
998     gluCylinder(Cylinder,.02,.02,.5,32,32);
999     glPopMatrix();
1000 }
1001 glPopMatrix();
1002 }
1003
1004 void house(void)
1005 {
1006     GLfloat mat_ambient[]={1,0,0,1};
1007     GLfloat mat_specular[]={1,1,1,1};
1008     GLfloat mat_diffuse[]={1,1,.7,1};
1009     GLfloat mat_shininess[]={50};
1010     matprop(mat_ambient,mat_diffuse,mat_specular,mat_shininess);
1011     GLfloat lightIntensity4[]={.7,.7,.7,.7};
1012     GLfloat light_position4[]={3,1,.5,1};
1013     glLightfv(GL_LIGHT6,GL_POSITION,light_position4);
1014     glLightfv(GL_LIGHT6,GL_DIFFUSE,lightIntensity4);
1015     glEnable(GL_LIGHT6);
1016     glPushMatrix();
1017     glTranslated(0,.15,0);
1018
1019     //roof
1020     glPushMatrix();
1021     glTranslated(-.02*4,3.9,-.01*4-.25);
1022     glScaled(1.5+.05,1.5,1.1);
1023     wall(0.08);
1024     glPopMatrix();
1025     GLfloat ambient2[]={1,0,0,1};
1026     GLfloat specular2[]={1,1,1,1};
1027     GLfloat diffuse2[]={.7,1,0.8,1};
1028     GLfloat shininess[]={50};
1029     matprop(ambient2,diffuse2,specular2,shininess);
1030
1031     //floor
1032     glPushMatrix();
1033     glTranslated(-.02*3,-0.05,-.01*4);
1034     glScaled(1.5+.01,1.5,1);
1035     wall(0.08);
1036     glPopMatrix();
1037
1038     GLfloat ambient1[]={1,0,0,1};
1039     GLfloat specular1[]={1,1,1,1};
1040     GLfloat diffusel[]={1,1,.7,1};
1041     GLfloat shininess1[]={50};
1042     matprop(ambient1,diffusel,specular1,shininess1);
1043     //left wall
1044     glPushMatrix();
1045     glRotated(90.0,0,0,1);
1046     wall(0.08);
1047     glPopMatrix();
1048     //right wall
1049     glPushMatrix();
1050     glTranslated(6,0,0);
1051     glRotated(90.0,0,0,1);
1052     wall(0.08);
1053     glPopMatrix();
1054     //back wall
1055     glPushMatrix();
1056     glTranslated(-.08,0,0);

```



```
1057 glScaled(1.5+.02,1,1);
1058 glRotated(-90.0,1,0,0);
1059 wall(0.08);
1060 glPopMatrix();
1061 //room vertical wall
1062 glPushMatrix();
1063 glTranslated(4,0,0);
1064 glScaled(1,1,.5);
1065 glRotated(90.0,0,0,1);
1066 wall(0.08);
1067 glPopMatrix();
1068 //room horizontal wall
1069 glPushMatrix();
1070 glTranslated(4.4,0,2);
1071 glScaled(.4,1,1);
1072 glRotated(-90.0,1,0,0);
1073 wall(0.08);
1074 glPopMatrix();
1075 //wall above the room door
1076 glPushMatrix();
1077 glTranslated(4,3,2);
1078 glScaled(.11,.25,1);
1079 glRotated(-90.0,1,0,0);
1080 wall(0.08);
1081 glPopMatrix();
1082 //left room horizontal wall
1083 glPushMatrix();
1084 glTranslated(0,0,2);
1085 glScaled(.4,1,1);
1086 glRotated(-90.0,1,0,0);
1087 wall(0.08);
1088 glPopMatrix();
1089 //lroom vertical wall
1090 glPushMatrix();
1091 glTranslated(1.6,0,0);
1092 glScaled(1,1,.35);
1093 glRotated(90.0,0,0,1);
1094 wall(0.08);
1095 glPopMatrix();
1096 //entrance room right wall
1097 glPushMatrix();
1098 glTranslated(1.6,0,2.59);
1099 glScaled(1,1,.35);
1100 glRotated(90.0,0,0,1);
1101 wall(0.08);
1102 glPopMatrix();
1103 //wall above main door
1104 glPushMatrix();
1105 glTranslated(-0.02,3,4);
1106 glScaled(.13,.27,1);
1107 glRotated(-90.0,1,0,0);
1108 wall(0.08);
1109 glPopMatrix();
1110 //wall right to the main door
1111 glPushMatrix();
1112 glTranslated(.48,0,4);
1113 glScaled(.68,1,1);
1114 glRotated(-90.0,1,0,0);
1115 wall(0.08);
1116 glPopMatrix();
1117 //wall right to the window
1118 glPushMatrix();
1119 glTranslated(4.8,0,4);
1120 glScaled(.3,1,1);
1121 glRotated(-90.0,1,0,0);
1122 wall(0.08);
```

```
1123 glPopMatrix();
1124 //wall below the window
1125 glPushMatrix();
1126 glTranslated(3.2,0,4);
1127 glScaled(.4,.25,1);
1128 glRotated(-90.0,1,0,0);
1129 wall(0.08);
1130 glPopMatrix();
1131 //wall above the window
1132 glPushMatrix();
1133 glTranslated(3.2,3.03,4);
1134 glScaled(.4,.25,1);
1135 glRotated(-90.0,1,0,0);
1136 wall(0.08);
1137 glPopMatrix();
1138 room();
1139 watertank();
1140 terece();
1141 steps();
1142 window();
1143 fan();
1144 cot(.6,.9,.06,.35,.009);
1145 diningtable();
1146 myclock();
1147 solar();
1148 GLfloat ambient[]={1,0.5,.5,1};
1149 GLfloat specular[]={1,1,1,1};
1150 GLfloat diffuse[]={1,.5,.5,1};
1151 matprop(ambient,diffuse,specular,mat_shininess);
1152 //main door
1153 glPushMatrix();
1154 glTranslated(0,0,4);
1155 glRotated(maino,0,1,0);
1156 glTranslated(0,0,-4);
1157 glPushMatrix();
1158 glTranslated(0,0,4);
1159 glScaled(.12,.75,1);
1160 glRotated(-90.0,1,0,0);
1161 wall(0.04);
1162 glPopMatrix();
1163 glPushMatrix();
1164 glTranslated(0,0,4);
1165 glScaled(.5,1,.2);
1166 glRotated(-90,1,0,0);
1167 gluCylinder(Cylinder, 0.05, 0.05, 3, 16, 16);
1168 glPopMatrix();
1169 glPopMatrix();
1170 //bolow room door
1171 glPushMatrix();
1172 glTranslated(4,0,(2-.025));
1173 glRotated(romo,0,1,0);
1174 glTranslated(-4,0,-(2-.025));
1175 glPushMatrix();
1176 glTranslated(4,0,2);
1177 glScaled(.099,.75,1);
1178 glRotated(-90.0,1,0,0);
1179 wall(0.01);
1180 glPopMatrix();
1181 glPushMatrix();
1182
1183 glTranslated(4.01,0,2-.025);
1184 glScaled(.5,1,.6);
1185 glRotated(-90,1,0,0);
1186 gluCylinder(Cylinder, 0.05, 0.05, 3, 16, 16);
1187 glPopMatrix();
1188 glPopMatrix();
```

```

1189     glPopMatrix();
1190     glFlush();
1191 }
1192 void display(void)
1193 {
1194     time(&ltime); // Get time
1195     newtime = localtime(&ltime); // Convert to local time
1196     glMatrixMode(GL_MODELVIEW);
1197     glLoadIdentity();
1198     glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT);
1199     gluLookAt(view[0],view[1],view[2],look[0],look[1],look[2],0.0,1.0,0.0);
1200     earth();
1201     compound();
1202     house();
1203     glFlush();
1204     glutSwapBuffers();
1205     glutPostRedisplay();
1206 }
1207 void Keyboard(unsigned char key,int x,int y)
1208 {
1209     switch(key)
1210     {
1211         //to move the camera along -ve x axis
1212         case '4':
1213             view[0]-=.1;
1214             glutPostRedisplay();
1215             break;
1216         //to move the camera along +ve x axis
1217         case '6':
1218             view[0]+=.1;
1219             glutPostRedisplay();
1220             break;
1221         //to move the camera along +ve y axis
1222         case '7':
1223             view[1]+=.1;
1224             glutPostRedisplay();
1225             break;
1226         //to move the camera along -ve y axis
1227         case '1':
1228             if(view[1]>1.9)
1229                 view[1]-=.1;
1230             glutPostRedisplay();
1231             break;
1232         //to move the camera along -ve z axis
1233         case '8':
1234             view[2]-=.1;
1235             glutPostRedisplay();
1236             break;
1237         //to move the camera along +ve z axis
1238         case '2':
1239             view[2]+=.1;
1240             glutPostRedisplay();
1241             break;
1242         //to run and stop the fan
1243         case 'S':
1244         case 's':
1245             flag*=-1;
1246             glutPostRedisplay();
1247             break;
1248         //to move the look position along +ve x axis
1249         case 'r':
1250         case 'R':
1251             look[0]+=.1;
1252             break;
1253         //to move the look position along -ve x axis
1254         case 'l':

```

```
1255 case 'L':
1256 look[0]-=.1;
1257 break;
1258 //to move the look position along +ve y axis
1259 case 'U':
1260 case 'u':
1261 look[1]+=.1;
1262 break;
1263 //to move the look position along -ve y axis
1264 case 'D':
1265 case 'd':
1266 look[1]-=.1;
1267 break;
1268 //to move the look position along +ve z axis
1269 case 'f':
1270 case 'F':
1271 look[2]+=.1;
1272 break;
1273 //to move the look position along -ve z axis
1274 case 'B':
1275 case 'b':
1276 look[2]-=.1;
1277 break;
1278 //to open and close the main door
1279 case 'q':
1280 case 'Q':
1281 if(maino==0)
1282     maino=85;
1283 else
1284     maino=0;
1285 break;
1286 //to open and close the below room door
1287 case 'O':
1288 case 'o':
1289 if(romo==0)
1290     romo=75;
1291 else
1292     romo=0;
1293 break;
1294 //to open and close the above room door
1295 case 'p':
1296 case 'P':
1297 if(tro==0)
1298     tro=70;
1299 else
1300     tro=0;
1301 break;
1302 //to open and close the main gate
1303 case 'g':
1304 case 'G':
1305 if(mgo==0)
1306     mgo=1;
1307 else
1308     mgo=0;
1309 break;
1310 //to open and close the sub gate
1311 case 'h':
1312 case 'H':
1313 if(sgo==0)
1314     sgo=50;
1315 else
1316     sgo=0;
1317 break;
1318 //inside view
1319 case 'i':
1320 case 'I':
```

```

1321 view[0]=2.8;
1322 view[1]=2;
1323 view[2]=4.8;
1324 look[0]=2.8;
1325 look[1]=2;
1326 look[2]=1;
1327 break;
1328 //top view
1329 case 'T':
1330 case 't':
1331 view[0]=6;
1332 view[1]=12;
1333 view[2]=10;
1334 look[0]=2;
1335 look[1]=4;
1336 look[2]=2;
1337 break;
1338 //front view
1339 case 'j':
1340 case 'J':
1341 view[0]=2;
1342 view[1]=2;
1343 view[2]=12.9;
1344 look[0]=3;
1345 look[1]=2;
1346 look[2]=3;
1347 break;
1348 //back view
1349 case 'k':
1350 case 'K':
1351     view[0]=1;
1352     view[1]=6;
1353     view[2]=-7;
1354     look[0]=2;
1355     look[1]=4;
1356     look[2]=2;
1357 break;
1358 }
1359 }
1360 void mySpecialKeyFunc( int key, int x, int y )
1361 {
1362     switch ( key ) {
1363     case GLUT_KEY_UP:
1364         if ( speed < 25.0 ) {
1365             speed+=5;
1366         }
1367         break;
1368     case GLUT_KEY_DOWN:
1369         if (speed>0) {
1370             speed-=5;
1371         }
1372         break;
1373     }
1374 }
1375 void main_menu(int m)
1376 {
1377     switch(m)
1378     {
1379     case 1:
1380         exit(0);
1381     }
1382 }
1383 void fan_menu(int m)
1384 {
1385     switch(m)
1386     {

```

```

1387     case 1:
1388         flag*=-1;
1389         glutPostRedisplay();
1390     break;
1391     case 2:
1392         if ( speed < 30.0)
1393             {
1394                 speed+=5;
1395             }
1396     break;
1397     case 3:
1398         if (speed>0)
1399             {
1400                 speed-=5;
1401             }
1402     break;
1403     }
1404 }
1405 void door_menu(int m)
1406 {
1407     switch(m)
1408     {
1409     case 1:
1410         if(maino==0)
1411             maino=85;
1412         else
1413             maino=0;
1414     break;
1415     case 2:
1416         if(romo==0)
1417             romo=75;
1418         else
1419             romo=0;
1420     break;
1421     case 3:
1422         if(tro==0)
1423             tro=90;
1424         else
1425             tro=0;
1426     break;
1427     }
1428 }
1429 void gate_menu(int m)
1430 {
1431     switch(m)
1432     {
1433     case 1:
1434         if(mgo==0)
1435             mgo=1;
1436         else
1437             mgo=0;
1438         break;
1439     case 2:
1440         if(sgo==0)
1441             sgo=50;
1442         else
1443             sgo=0;
1444         break;
1445     }
1446 }
1447 void house_view(int m)
1448 {
1449     switch(m)
1450     {
1451     case 1:
1452

```

```

1453     view[0]=2.8;
1454     view[1]=2;
1455     view[2]=4.8;
1456     look[0]=2.8;
1457     look[1]=2;
1458     look[2]=1;
1459     break;
1460     case 2:
1461         view[0]=6;
1462         view[1]=12;
1463         view[2]=10;
1464         look[0]=2;
1465         look[1]=8;
1466         look[2]=2;
1467     break;
1468     case 3:
1469         view[0]=2;
1470         view[1]=2;
1471         view[2]=12.9;
1472         look[0]=3;
1473         look[1]=2;
1474         look[2]=3;
1475     break;
1476     case 4:
1477         view[0]=1;
1478         view[1]=6;
1479         view[2]=-7;
1480         look[0]=2;
1481         look[1]=4;
1482         look[2]=2;
1483     break;
1484 }
1485 }
1486 void menu()
1487 {
1488     int sub_menu1=glutCreateMenu(fan_menu);
1489     glutAddMenuEntry("on/off fan(s)",1);
1490     glutAddMenuEntry("speed up(up arrow)",2);
1491     glutAddMenuEntry("speed down(down arrow)",3);
1492
1493     int sub_menu2=glutCreateMenu(door_menu);
1494     glutAddMenuEntry("main door(q)",1);
1495     glutAddMenuEntry("ground floor room door(o)",2);
1496     glutAddMenuEntry("1st floor room door(p)",3);
1497
1498
1499     int sub_menu3=glutCreateMenu(gate_menu);
1500     glutAddMenuEntry("main gate(g)",1);
1501     glutAddMenuEntry("sub gate(h)",2);
1502
1503     int sub_menu4=glutCreateMenu(house_view);
1504     glutAddMenuEntry("front view(j)",3);
1505     glutAddMenuEntry("top view(t)",2);
1506     glutAddMenuEntry("inside view(i)",1);
1507     glutAddMenuEntry("back view(k)",4);
1508
1509     glutCreateMenu(main_menu);
1510     glutAddMenuEntry("Quit",1);
1511     glutAddSubMenu("fan menu",sub_menu1);
1512     glutAddSubMenu("open/close door",sub_menu2);
1513     glutAddSubMenu("open/close gate",sub_menu3);
1514     glutAddSubMenu("house view",sub_menu4);
1515     glutAttachMenu(GLUT_LEFT_BUTTON);
1516
1517 }
1518 int main(int argc,char**argv)

```

```
1519 {
1520     glutInit(&argc,argv); //to initialize the glut library
1521     glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);
1522     glutInitWindowSize(w,h);
1523     glutInitWindowPosition(0,0);
1524     glutCreateWindow("3D-HOUSE");
1525     myinit();
1526     glutDisplayFunc(display);
1527     glutKeyboardFunc(Keyboard);
1528     glutSpecialFunc(mySpecialKeyFunc);
1529     menu();
1530     //glutFullScreen(); //to see o/p in full screen on monitor
1531     glEnable(GL_LIGHTING);
1532     glEnable(GL_LIGHT0);
1533     glShadeModel(GL_SMOOTH); //smooth shaded
1534     glEnable(GL_DEPTH_TEST); //to remove hidden surface
1535     glEnable(GL_NORMALIZE); //to make normal vector to unit vector
1536     glClearColor(0,.3,.8,0);
1537     glViewport(0,0,w,h);
1538     glutMainLoop();
1539     //return 0;
1540 }
```