CP2 Exercise 15 **19/02/2007**

1. Create a GLUT program that draws a solar system (see example in Red Book). The solar system should have a sun and two planets (each rotating around the sun and about its own axis). One of the planets should have a single moon, and the other should have two moons, rotating around their respective planets.
Use double buffering and the Linux high resolution timer to achieve smooth animation of the stellar objects.

note: *the function **glutPostRedisplay()** forces the scene to be redrawn.*

```
#include <GL/glut.h>
#include <sys/time.h> /* Linux system clock functions */
#include <math.h>

double framestart=-1.0,frameend=0.0;
double frametime() /* request frame time from Linux */
{
  struct timeval tv;
  struct timezone tz;
  gettimeofday(&tv, &tz);
  return (double)tv.tv_sec + (double)tv.tv_usec/(1000*1000);
}

void idle(void)
{
  glutPostRedisplay();
}

void display(void) /* display callback */
{
  /* frame time management */
  double timer;
  static double rotation=0.0;

  if(framestart<0.0) framestart=frametime();
  do{
    frameend=frametime();
  }while(frameend==framestart);

  timer=frameend-framestart;
  framestart=frameend;

  /* rendering code goes here */

  glutSwapBuffers();
}
```

2. Write a small GLUT program that draws a simple representation of a car (*2 boxes, 4 tyres*). Create different versions of the program that use vertex arrays/display lists for drawing the objects.

hint: *You will have to construct the tyres from combinations of different primitives.*

3. Encapsulate the bitmaps for the characters of the font described below within OpenGL display lists.

a) Write a simple GLUT program that accepts a character string (*maximum length - 50 characters*) from the shell and then prints that string into the GLUT OpenGL window, using the font display lists.

b) Expand the program by adding a function **`printg()`** with a variable parameter-list which takes similar parameters to the ANSI C ***printf*** function but which does not print its output to the standard output (*shell*), but instead uses the defined font for printing to the GLUT OpenGL window.

```
GLubyte space[] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                   0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}; /*  */
GLubyte colon[] = {0x00, 0x00, 0x00, 0x00, 0x018, 0x18,
                   0x00, 0x00, 0x18, 0x18, 0x00, 0x00, 0x00}; /* : */
GLubyte exclamation[] = {0x00, 0x00, 0x18, 0x18, 0x00,
                   0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18}; /* ! */
GLubyte period[] = {0x00, 0x00, 0x18, 0x18, 0x00, 0x00,
                    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}; /* . */

GLubyte numbers[][13] = {
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xe7, 0x7e}, /* 0 */
{0x00, 0x00, 0x7e, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x38, 0x18}, /* 1 */
{0x00, 0x00, 0xff, 0xe0, 0x60, 0x30, 0x18, 0x0c, 0x06, 0x03, 0x03, 0xe7, 0x7e}, /* 2 */
{0x00, 0x00, 0x7e, 0xe7, 0x03, 0x03, 0x07, 0x0e, 0x07, 0x03, 0x03, 0xe7, 0x7e}, /* 3 */
{0x00, 0x00, 0x06, 0x06, 0xff, 0xc6, 0xc6, 0xc6, 0x66, 0x36, 0x1e, 0x0e, 0x06}, /* 4 */
{0x00, 0x00, 0x7e, 0xe7, 0x03, 0x03, 0x03, 0x07, 0x7e, 0xc0, 0xc0, 0xc0, 0xff}, /* 5 */
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xc3, 0xc7, 0xfe, 0xc0, 0xc0, 0xe7, 0x7e}, /* 6 */
{0x00, 0x00, 0x30, 0x30, 0x30, 0x30, 0x30, 0x18, 0x0c, 0x06, 0x03, 0xc3, 0xff}, /* 7 */
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xe7, 0x7e, 0xe7, 0xc3, 0xc3, 0xe7, 0x7e}, /* 8 */
{0x00, 0x00, 0x7e, 0xe7, 0x03, 0x03, 0x03, 0x7f, 0xe3, 0xc3, 0xc3, 0xe7, 0x7e}  /* 9 */
};

GLubyte capitals[][13] = {
{0x00, 0x00, 0xc3, 0xc3, 0xc3, 0xc3, 0xff, 0xc3, 0xc3, 0xc3, 0xc3, 0xe7, 0x7e}, /* A */
{0x00, 0x00, 0xfe, 0xc7, 0xc3, 0xc3, 0xc7, 0xfe, 0xc7, 0xc3, 0xc3, 0xc7, 0xfe}, /* B */
{0x00, 0x00, 0x7e, 0xe7, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xe7, 0x7e}, /* C */
{0x00, 0x00, 0xfe, 0xc7, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc7, 0xfe}, /* D */
{0x00, 0x00, 0xff, 0xc0, 0xc0, 0xc0, 0xc0, 0xfc, 0xc0, 0xc0, 0xc0, 0xc0, 0xff}, /* E */
{0x00, 0x00, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xfc, 0xc0, 0xc0, 0xc0, 0xff}, /* F */
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xcf, 0xc0, 0xc0, 0xc0, 0xc0, 0xe7, 0x7e}, /* G */
{0x00, 0x00, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xff, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3}, /* H */
{0x00, 0x00, 0x7e, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x7e}, /* I */
{0x00, 0x00, 0x7c, 0xee, 0xc6, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06}, /* J */
{0x00, 0x00, 0xc3, 0xc6, 0xcc, 0xd8, 0xf0, 0xe0, 0xf0, 0xd8, 0xcc, 0xc6, 0xc3}, /* K */
{0x00, 0x00, 0xff, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0}, /* L */
{0x00, 0x00, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xdb, 0xff, 0xff, 0xe7, 0xc3}, /* M */
{0x00, 0x00, 0xc7, 0xc7, 0xcf, 0xcf, 0xdf, 0xdb, 0xfb, 0xf3, 0xf3, 0xe3, 0xe3}, /* N */
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xe7, 0x7e}, /* O */
{0x00, 0x00, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xfe, 0xc7, 0xc3, 0xc3, 0xc7, 0xfe}, /* P */
{0x00, 0x00, 0x7f, 0xee, 0xdf, 0xdb, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xe7, 0x7e}, /* Q */
{0x00, 0x00, 0xc3, 0xc6, 0xcc, 0xd8, 0xf0, 0xfe, 0xc7, 0xc3, 0xc3, 0xc7, 0xfe}, /* R */
{0x00, 0x00, 0x7e, 0xe7, 0x03, 0x03, 0x07, 0x7e, 0xe0, 0xc0, 0xc0, 0xe7, 0x7e}, /* S */
{0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0xff}, /* T */
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3}, /* U */
{0x00, 0x00, 0x18, 0x3c, 0x3c, 0x66, 0x66, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3}, /* V */
{0x00, 0x00, 0xc3, 0xe7, 0xff, 0xff, 0xdb, 0xdb, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3}, /* W */
{0x00, 0x00, 0xc3, 0x66, 0x66, 0x3c, 0x3c, 0x18, 0x3c, 0x3c, 0x66, 0x66, 0xc3}, /* X */
{0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x3c, 0x3c, 0x66, 0x66, 0xc3}, /* Y */
{0x00, 0x00, 0xff, 0xc0, 0xc0, 0x60, 0x30, 0x7e, 0x0c, 0x06, 0x03, 0x03, 0xff}  /* Z */
};

GLubyte lowercase[][13] = {
{0x00, 0x00, 0x7f, 0xe7, 0xc3, 0xc3, 0xc3, 0xe7, 0x7f, 0x00, 0x00, 0x00, 0x00}, /* a */
{0x00, 0x00, 0xfe, 0xe7, 0xc3, 0xc3, 0xc3, 0xe7, 0xfe, 0xc0, 0xc0, 0xc0, 0x00}, /* b */
{0x00, 0x00, 0x7e, 0xe7, 0xc0, 0xc0, 0xc0, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* c */
{0x00, 0x00, 0x7f, 0xe7, 0xc3, 0xc3, 0xc3, 0xe7, 0x7f, 0x03, 0x03, 0x03, 0x00}, /* d */
{0x00, 0x00, 0x7e, 0xe7, 0xc0, 0xff, 0xc3, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* e */
{0x00, 0x00, 0x30, 0x30, 0x30, 0x30, 0x30, 0x7c, 0x30, 0x30, 0x30, 0x1c, 0x00}, /* f */
{0x7e, 0xe7, 0x03, 0x7f, 0xe7, 0xc3, 0xc3, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* g */
{0x00, 0x00, 0xc6, 0xc6, 0xc6, 0xc6, 0xc6, 0xee, 0xfc, 0xc0, 0xc0, 0xc0, 0x00}, /* h */
{0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x00, 0x00, 0x18, 0x18, 0x00}, /* i */
{0x30, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x00, 0x00, 0x18, 0x18, 0x00}, /* j */
{0x00, 0x00, 0xc6, 0xce, 0xdc, 0xf8, 0xf0, 0xf8, 0xdc, 0xce, 0xc6, 0xc0, 0x00}, /* k */
{0x00, 0x00, 0x0c, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x00}, /* l */
{0x00, 0x00, 0xdb, 0xdb, 0xdb, 0xdb, 0xdb, 0xff, 0xda, 0x00, 0x00, 0x00, 0x00}, /* m */
{0x00, 0x00, 0xc6, 0xc6, 0xc6, 0xc6, 0xc6, 0xee, 0xfc, 0x00, 0x00, 0x00, 0x00}, /* n */
{0x00, 0x00, 0x7e, 0xe7, 0xc3, 0xc3, 0xc3, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* o */
{0xc0, 0xc0, 0xc0, 0xfe, 0xe7, 0xc3, 0xc3, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* p */
{0x03, 0x03, 0x03, 0x7f, 0xe7, 0xc3, 0xc3, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* q */
{0x00, 0x00, 0xc0, 0xc0, 0xc0, 0xc0, 0xc0, 0xe0, 0xfc, 0x00, 0x00, 0x00, 0x00}, /* r */
{0x00, 0x00, 0x7e, 0xe7, 0x03, 0xff, 0xc0, 0xe7, 0x7e, 0x00, 0x00, 0x00, 0x00}, /* s */
{0x00, 0x00, 0x1c, 0x30, 0x30, 0x30, 0x30, 0x7c, 0x30, 0x30, 0x30, 0x30, 0x00}, /* t */
{0x00, 0x00, 0x7f, 0xe7, 0xc3, 0xc3, 0xc3, 0xc3, 0xc3, 0x00, 0x00, 0x00, 0x00}, /* u */
{0x00, 0x00, 0x18, 0x3c, 0x3c, 0x66, 0x66, 0xc3, 0xc3, 0x00, 0x00, 0x00, 0x00}, /* v */
{0x00, 0x00, 0xe7, 0xff, 0xff, 0xdb, 0xdb, 0xdb, 0xc3, 0x00, 0x00, 0x00, 0x00}, /* w */
{0x00, 0x00, 0xc6, 0xee, 0x7c, 0x38, 0x7c, 0xee, 0xc6, 0x00, 0x00, 0x00, 0x00}, /* x */
{0xe0, 0x70, 0x38, 0x3c, 0x3c, 0x66, 0x66, 0xc3, 0xc3, 0x00, 0x00, 0x00, 0x00}, /* y */
{0x00, 0x00, 0xfe, 0xe0, 0x70, 0x38, 0x1c, 0x0e, 0xfe, 0x00, 0x00, 0x00, 0x00}  /* z */
};

GLuint fontOffset;
```