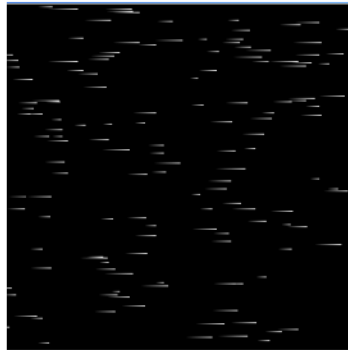


SORU:

SORU 5) Yıldız yağmurunu (yaklaşık 150 adet yıldız) canlandıran şekildeki gibi bir görüntü veren bir cpp uygulamasını yazınız. Burada her bir yıldız `GL_LINE_STRIP` kullanılarak oluşturulmalı ve zaman içinde soldan sağa doğru tek bir doğrultuda hareket etmelidir. Programda ayrıca kullanıcı "f" tuşuna bastığında `glutFullScreen()` devreye girmeli, "w" tuşuna basınca pencere eski konumuna dönmelidir.



CEVAP:

Tanımlamalar

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <GL/glut.h>
```

star adında iki alt değişkene sahip yapı tanımlanmıştır.

```
typedef struct _star {
    float x, y;
    float vx, vy;
} star;

int k_width = 4; //yıldız uzunluk katsayı değişkeni
star* stars = NULL; //star tipi stars adında değişken tanımlı
int num_stars = 150; //maksimum yıldız sayısını tutan değişken
```

```

void
reshape(int width, int height)
{
    int i = 0;
    glViewport(0, 0, width, height);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(0, width, 0, height, -1, 1);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    glColor3ub(255, 255, 255);

```

stars değişkenine döngü içerisinde ekran yükseklik ve genişliğinde random sayı üretilerek x ve y alt değişkenlerine yazılmıştır.

```

    for (i = 0; i < num_stars; i++) {
        stars[i].x = rand() % width;
        stars[i].y = rand() % height;

```

x ekseninde kuyruk oluşturulacağından random maksimum değeri olan 32767 ile 5 sayısı çarpılıp 2 eklenerek vx alt değişkenine yazılıyor.

```

        stars[i].vx = rand() / (float)RAND_MAX * 5 + 2;

```

y eksenini sabit olacağından vy alt değişkenine 0 yazılıyor.

```

        stars[i].vy = 0;
    }
}

```

display fonksiyonu

```

Void display(void)
{
    int i;
    glClear(GL_COLOR_BUFFER_BIT);

    for (i = 0; i < num_stars; i++) {

```

```
stars[i].x += stars[i].vx;
```

üretile x sayısı oluşturulan pencere genişliğinden küçük olup olmadığı kontrol

```
if (stars[i].x < glutGet(GLUT_WINDOW_WIDTH)) {  
    glBegin(GL_LINE_STRIP);  
    glColor3f(0, 0, 0);
```

x değerinden vx değerinin k_width katı kadar eksiği ilk vertex in x değeri olarak tanımlanıyor.
y değeri daha önce random üretilen sayı vertexin y değeri olarak tanımlanıyor

```
    glVertex2i(stars[i].x-stars[i].vx*k_width,  
stars[i].y);  
    glColor3f(255, 255, 255);
```

diğer noktası x ve y alt değişkende olan sayılar tanımlanıyor.

```
    glVertex2i(stars[i].x, stars[i].y);  
    glEnd();  
} else {  
    stars[i].x = 0; //pencere dışındaysa x değişkenine 0 değeri  
veriliyor.  
}  
}  
glutSwapBuffers();  
}
```

program boşta durdukça yeniden display yapmaktadır.

```
void idle(void)  
{  
    glutPostRedisplay();  
}
```

klavye tuş fonksiyonları

```
void keyboard(unsigned char key, int x, int y)  
{
```

İlk ekran ölçüleri tanımlamaları

```
static int old_x = 50;  
static int old_y = 50;
```

```
static int old_width = 320;
static int old_height = 320;
switch (key) {
```

w tuşu ile ekran boyutu ilk haline getiriliyor.

```
case 'w':
    glutPositionWindow(old_x, old_y);
    glutReshapeWindow(old_width, old_height);
    break;
```

f tuşu ile ekran boyutu GLUT_WINDOW fonksiyonundan dönen değerlere eşitlenmektedir.

```
case 'f':
    if (glutGet(GLUT_WINDOW_WIDTH) <
        glutGet(GLUT_SCREEN_WIDTH)) {
        old_x = glutGet(GLUT_WINDOW_X);
        old_y = glutGet(GLUT_WINDOW_Y);
        old_width = glutGet(GLUT_WINDOW_WIDTH);
        old_height = glutGet(GLUT_WINDOW_HEIGHT);
        glutFullScreen();
    }
    break;
}
}
```

```
int main(int argc, char** argv)
{
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA);
    glutInitWindowPosition(50, 50);
    glutInitWindowSize(320, 320);
    glutInit(&argc, argv);
    glutCreateWindow("soru5");
    glutDisplayFunc(display);
}
```

```
glutReshapeFunc(reshape);  
glutKeyboardFunc(keyboard);  
if (argc > 1) {  
    if (strcmp(argv[1], "-h") == 0) {  
        fprintf(stderr, "%s [stars]\n", argv[0]);  
        exit(0);  
    }  
    sscanf(argv[1], "%d", &num_stars);  
}  
stars = (star*)malloc(sizeof(star) * num_stars);  
glutIdleFunc(idle);  
glutMainLoop();  
return 0;  
}
```

Ek

finalsor5.cpp

finalsor5.exe

Ekran Çıktısı

