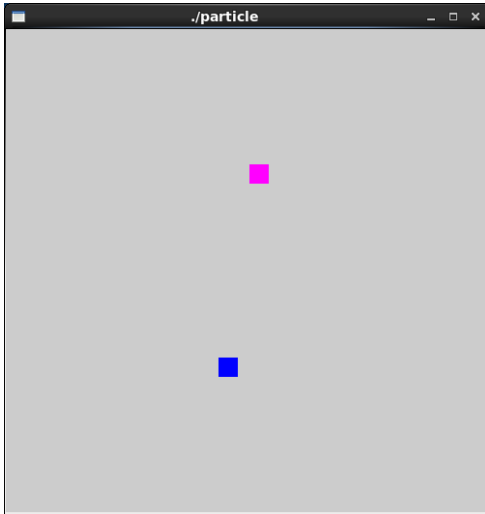


*Write a shader program that continuously shoots a particle from the left side of the screen to the right side and another particle from the right to the left.*



Code:

//particle.cpp

...

void display(void)

{

GLfloat vec[4];

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glClearColor( 0.8, 0.8, 0.8, 0.0 ); //get white background color

glColor3f ( 1, 0, 0 ); //red, this will have no effect if shader is loaded

glPointSize ( 20 );

//"shoot" a particle at 45 degrees

glBegin ( GL\_POINTS ); //need "GL\_POINTS"; "GL\_POINT" does not work

glVertexAttrib3f ( velParam, 10, 10, 0 ); //send vel to vertex shader

glVertexAttrib3f( loc, 1, 0, 1 );

glVertex2f ( -15, 6 ); //starting position of particle

glEnd();

glBegin( GL\_POINTS );

glVertexAttrib3f( velParam, -10, 10, 0 );

glVertexAttrib3f( loc, 0, 0, 1 );

glVertex2f( 15, -6 );

glEnd();

glutSwapBuffers();

```

    glFlush();
}
...

//particle.vert

...
void main(void)
{
    color = temp;

    float s = 1000.0;           //scale factor
    float g = -10.0;
    float t;
    t = time / s;              //time in ms
    vec4 object_pos = gl_Vertex; //starting position

    object_pos.x = object_pos.x + vel.x*t;
    object_pos.y = object_pos.y;
    object_pos.z = object_pos.z + vel.z*t;

    gl_Position = gl_ModelViewProjectionMatrix * object_pos;
}

```

Report:

I have successfully finished all parts of lab 7.