# BCA613 Computer Animation I

Hazırlayan Tayfun GÜRLEVİK N19139647

12.04.2020

## ÖDEV3)

3B uzay içinde verilen bir poligon ve bir nokta verilmektedir. Verilen noktanın poligonun içinde mi yoksa dışında mı olduğunu veren algoritmayı öğrenip (5 hafta yansıları içinde 10.-15. yansılar) karşı gelen programı yazınız.

### Çözüm:

Çözüm için C++ proglama dili ve OpenGL grafik kütüphanesi kullanılmıştır.

Point isminde bir struct, Segment ve Poligon isminde class lar oluşturulmuştur.

Poligonun nesnesinin noktaları aşağıdaki kod parçası vasıtasıyla oluşturulmuştur.

for (int i = 0; i <4; i++)

{

Point p;

if (i%3==0)

{

p.position.x = i \* 4;

p.position.y= i \* i \* 2;

p.position.z = 0;

}

else

{

p.position.x = i \* 7;

p.position.y = -i \* i - 5;

p.position.z = 0;

}

poligon.points.push\_back(p);

}

Mouse sol tuşu ile q noktası ekranda işaretlenerek, q noktasının poligonun içinde kalıp kalmadığı aşağıdaki kod parçası ile çözülmüştür.

if (button == GLUT\_LEFT\_BUTTON && state == GLUT\_UP)

{

std::cout << "q noktasinin koordinatlari:" << std::endl;

std::cout << "X: " << q.position.x << " Y: " << q.position.y << std::endl;

rays.clear();

zValues.clear();

for (size\_t i = 0; i < poligon.points.size(); i++)

{

glm::vec3 rayfrompoints;

rayfrompoints = q.position - poligon.points.at(i).position;

rays.push\_back(rayfrompoints);

}

for (size\_t i = 0; i < poligon.points.size(); i++)

{

zValues.push\_back( glm::cross(poligon.segments.at(i).edgeVector, rays.at(i)).z);

}

std::cout << "Vektorel carpimlarin z bilesenleri:" << std::endl;

int carpim=0;

bool isaretDegisti = false;

for (size\_t i = 0; i < zValues.size(); i++)

{

std::cout << zValues.at(i) << std::endl;

if (i > 0&&!isaretDegisti)

{

carpim = zValues.at(i) \* zValues.at(i - 1);

if (carpim<0)

{

isaretDegisti = true;

}

}

}

if (isaretDegisti)

{

std::cout << "q noktasi disarida" << std::endl;

}

else

{

std::cout << "q noktasi iceride" << std::endl;

}

}

### Ekler:

Odev3.cpp ve Odev3.mp4