**기초 컴퓨터 그래픽스**

**HW1 README**

20181210 유창호

**1. 개발환경**

|  |  |
| --- | --- |
| **운영체제** | **Windows 11 64bit** |
| **IDE** | **Visual Studio 2022** |
| **CPU** | **Intel i7-8570H** |

**2. 요구사항**

**(a) 윈도우 화면 구현**

구현 여부 : 예

작동 확인 방법 : 프로그램을 실행하면 윈도우가 화면에 뜬다.

구현 방법 : glutInitWindowSize(750,750)으로 750 X 750 크기의 윈도우 크기를 지정. glutInitWindowPosition(500, 200)를 사용해 500, 200 좌표에 윈도우를 위치하게 함.

**(b) 선분 그리기**

구현 여부 : 예

작동 확인 방법 : 프로그램을 실행하면 선분이 화면에 뜬다.

구현 방법 : initialize\_renderer 함수에서 선분의 양 점 px,py qx,py를 설정했다. 그리고 draw\_line 함수를 정의해 glBegin(GL\_LINES) 블록에서 선을 그리고 glBegin(GL\_POINTS) 블록에서는 점을 그렸다. glColor3f 함수를 이용해 양 점의 색갈을 흰색과 파란색으로 설정하고 glPointSize로 점의 크기를 지정했다.

**(c) 비대칭 다각형 그리기**

구현 여부 : 예

작동 확인 방법 : 프로그램을 실행하면 다각형이 화면에 뜬다.

구현 방법 : initialize\_renderer 함수에서 object 배열에서 다각형의 점들의 위치를 입력했다. 그리고 draw\_object 함수를 정의해 glBegin(GL\_LINE\_LOOP) 블록에서 점들을 잇는 선을 그리고 glBegin(GL\_POINTS) 블록에서 점들의 꼭짓점을 그렸다. glColor3f 함수를 이용해 꼭짓점들은 휜색으로 하고 무게중심을 파란색으로 했다 glPointSize로 점의 크기를 지정했다.

**(d) 선분 회전 구현 Rotation**

구현 여부 : 예

작동 확인 방법 : 마우스 휠을 사용해 선분을 왼쪽, 오른쪽으로 돌릴 수 있다.

구현 방법 : mousepress 함수의 if else 문에서 마우스의 휠을 위로 할 때(button == 3)와 아래로 할 때(button == 4)의 경우를 정의했고 각각의 경우에서 선분이 돌려지는 각도인 rotation\_angle\_in\_degree 변수의 값을 양수와 음수로 설정했다. 이후 rotate\_line 함수를 정의해서 이동하기 전 선분의 꼭짓점 b\_qx, b\_qy에 대해서 rotation\_angle\_in\_degree 값만큼 rotate한 rotation 행렬을 계산해서 새로운 꼭짓점의 위치 qx, qy를 결정했다.

**(e) SHIFT로 파란색 꼭지점 이동 Picking**

구현 여부 : 예

작동 확인 방법 : SHIFT를 누른 상태에서 파란점을 왼쪽 클릭해서 이동할 수 있다.

구현 방법 : special 콜백함수에서 GLUT\_KEY\_SHIFT\_L가 눌렸을 때 shiftpressed 변수를 1로 설정해 shift가 눌렸을 때를 감지한다. 그리고 mousepress 콜백함수에서 왼쪽 클릭을 감지했을 때 클릭한 위치를 glut 좌표로 변환해 clicked\_x\_glut, clicked\_y\_glut 변수에 저장한다. 그리고 shiftpressed의 변수가 1인 경우에는 clicked\_x\_glut, clicked\_y\_glut 와 파란색 꼭짓점인 px, py의 거리가 0.04f 안쪽을 만족한다면 해당 점을 클릭했다고 인식해 vertex\_clicked 변수를 1로 만든다. mousemove 함수에서 vertex\_clicked가 1이라면 해당 점을 마우스로 이동 중인 위치인 moveto\_glutx, moveto\_gluty 로 이동시키도록 정의한 move\_vertex함수를 호출한다. shift가 때졌을 때 shiftpressed 변수를 0으로 해서 기능을 멈추게 한다.

**(f) ALT로 다각형 이동 Translation**

구현 여부 : 예

작동 확인 방법 : ALT를 누른 상태에서 다각형을 오른쪽 클릭해서 이동할 수 있다.

구현 방법 : special 콜백함수에서 GLUT\_KEY\_ALT\_L가 눌렸을 때 altpressed 변수를 1로 설정해 alt가 눌렸을 때를 감지한다. 그리고 mousepress 콜백함수에서 오른쪽 클릭을 감지했을 때 클릭한 위치를 glut 좌표로 변환해 prev\_x\_glut, prev\_y\_glut 변수에 저장한다. 프레임마다 마우스가 이동한 거리를 mousemove 함수에서 계산한다. 그 전 프레임의 마우스의 위치가 prev\_x\_glut, prev\_y\_glut 좌표라고 한다면, mousemove 함수의 매개변수로 받는 x, y 값은 현재 마우스 위치를 나타내기 때문에 이 둘의 차이가 한 프레임 동안 마우스가 이동한 거리라고 계산할 수 있다. altpressed가 1일 때 move\_object 함수를 호출해서 translation 행렬을 계산해 해당 거리만큼 물체를 이동시킨다. alt을 누르지 않을 경우 unpressed 콜백함수에서 altpressed 값을 0으로 만들어 제어한다.

**(g) CTRL로 다각형 크기 조절 Scaling**

구현 여부 : 예

작동 확인 방법 : CTRL를 누른 상태에서 다각형을 오른쪽 클릭해 x 축 방향으로 이동시켜 크기를 조절할 수 있다.

구현 방법 : special 콜백함수에서 GLUT\_KEY\_CTRL\_L가 눌렸을 때 ctrlpressed 변수를 1로 설정해 ctrl가 눌렸을 때를 감지한다. 그리고 mousepress 콜백함수에서 오른쪽 클릭을 감지했을 때 클릭한 위치를 glut 좌표로 변환해 prev\_x\_glut, prev\_y\_glut 변수에 저장한다. 프레임마다 마우스가 이동한 거리를 mousemove 함수에서 계산해 scaling 정도를 나타내는 scale\_move 값을 계산한다. 이후 scale\_object 함수를 호출해 마우스의 x 좌표의 새로운 프레임에서의 위치가 전 프레임 보다 증가했다면 증가한 만큼 scale\_move 값을 더해주고, 감소했다면 감소한 만큼 scale\_move를 조절한다. 마우스가 양의 방향으로 이동한다면 scale\_move 값이 1 이상이 되어 도형이 커지게 하고 음의 방향으로 이동한다면 scale\_move 값이 1 이하가 되게 해서 작아지게 한다. ctrl을 누르지 않을 경우 unpressed 콜백함수에서 ctrlpressed 값을 0으로 만들어 제어한다.

**(h) 추가 구현 – 화살표 모양의 다각형 Shearing**

구현 여부 : 예

작동 확인 방법 : F1을 누른 상태에서 오른쪽 클릭을 한 다음 마우스를 이동하면 x 방향으로 shearing을 하고 F2를 누른 상태에서 오른쪽 클릭을 한 다음 마우스를 이동하면 y 방향으로 shearing을 한다.

구현 방법 :

**x 방향 shearing (F1)**

special 콜백함수에서 GLUT\_KEY\_F1가 눌렸을 때 f1pressed 변수를 1로 설정해 f1이 눌렸을 때를 감지한다. 그리고 mousepress 콜백함수에서 오른쪽 클릭을 감지했을 때 클릭한 위치를 glut 좌표로 변환해 prev\_x\_glut, prev\_y\_glut 변수에 저장한다. 프레임마다 마우스가 이동한 거리를 mousemove 함수에서 계산해 정도를 나타내는 shear\_move\_x 값을 계산한다. 이후 shear\_object\_x 함수를 호출해 x 방향으로 shearing을 적용하는 행렬식을 계산해 물체의 새로운 모양을 결정한다. f1을 누르지 않을 경우 unpressed 콜백함수에서 f1pressed 값을 0으로 만들어 제어한다.

**y 방향 shearing (F2)**

special 콜백함수에서 GLUT\_KEY\_F2가 눌렸을 때 f2pressed 변수를 1로 설정해 f2가 눌렸을 때를 감지한다. 그리고 mousepress 콜백함수에서 오른쪽 클릭을 감지했을 때 클릭한 위치를 glut 좌표로 변환해 prev\_x\_glut, prev\_y\_glut 변수에 저장한다. 프레임마다 마우스가 이동한 거리를 mousemove 함수에서 계산해 정도를 나타내는 shear\_move\_y 값을 계산한다. 이후 shear\_object\_y 함수를 호출해 y 방향으로 shearing을 적용하는 행렬식을 계산해 물체의 새로운 모양을 결정한다. f2을 누르지 않을 경우 unpressed 콜백함수에서 f2pressed 값을 0으로 만들어 제어한다.