기초 컴퓨터 그래픽스

HW1: GLUT 툴킷 사용 보고서

담당교수: 임인성

학번: 20231632

이름: Jumagul Alua

**작성 환경:**

Windows 10 64bit, i9-10980XE, RTX 4090, Visual Studio 2022 Release x64

**구현 기능과 구현 방법:**

a) 다각형 생성

확인 방법 :

* p 키를 눌러 다각형 생성 모드로 진입한다.
* CTRL 키를 누른 상태에서 오른쪽 마우스 버튼을 클릭하여 꼭지점을 추가한다.
* 최소 3개 이상의 점을 추가하면, 점들이 자동으로 연결되어 선이 그려진다.
* 다시 p 키를 누르면 다각형이 완성되고 무게중심이 계산된다.

구현 방법 :

* keyboard 함수에서 p 키 입력 시 다각형 생성 모드를 제어한다.
* mousepress 함수에서 GLUT\_RIGHT\_BUTTON과 GLUT\_ACTIVE\_CTRL 조합을 사용하여 점을 추가한다.
* add\_point 함수를 호출하여 점의 좌표를 계산하고 저장한다.
* close\_line\_segments와 update\_center\_of\_gravity 함수를 사용하여 다각형을 완성하고 무게중심을 계산한다.

b) 무게중심 점

확인 방법 :

* p 키를 눌러 다각형 생성이 완료된 후, 빨간색 중심점이 화면에 표시된다.

구현 방법 :

* update\_center\_of\_gravity 함수를 사용하여 다각형의 모든 점의 평균 좌표를 계산하여 무게중심을 결정한다.
* draw\_lines\_by\_points 함수에서 polygon\_mode가 활성화된 경우에만 중심점을 빨간색(CENTER\_POINT\_COLOR)으로 표시한다.

c) 선 색깔 변함

확인 방법 :

* 다각형 생성이 완료된 상태에서 빨간색 중심점을 클릭한다.
* 다각형 선의 색상이 파란색에서 핑크색으로 변경된다.

구현 방법 :

* mousepress 함수에서 중심점 클릭 여부를 계산한다.
* 클릭 거리가 임계값(0.03f) 이내일 경우 st.move\_mode를 활성화하고, line\_color를 핑크색(MOVE\_MODE\_COLOR)으로 변경한다.
* draw\_lines\_by\_points 함수에서 line\_color를 사용하여 선의 색상을 설정한다.

d) 중심점과 다각형 이동

확인 방법 :

* 빨간색 중심점을 클릭한 상태에서 마우스를 움직인다.
* 다각형과 중심점이 함께 이동한다.

구현 방법 :

* mousemove 함수에서 마우스 이동 거리를 계산하고, 이를 move\_points 함수를 통해 다각형의 모든 점에 적용한다.
* 중심점 좌표(pg.center\_x, pg.center\_y)도 동일한 이동 거리만큼 업데이트한다.
* glutPostRedisplay를 호출하여 화면을 새로 고친다.

e) 선분의 색깔이 원래대로 돌아가기

확인 방법 :

* 빨간색 중심점을 클릭한 상태에서 마우스 버튼을 놓는다.
* 다각형 선의 색상이 핑크색에서 파란색으로 돌아간다.

구현 방법 :

* mousepress 함수에서 마우스 버튼이 놓였을 때(state == GLUT\_UP) st.move\_mode를 비활성화한다.
* line\_color를 기본값인 파란색(LINE\_COLOR)으로 복원한다.
* glutPostRedisplay를 호출하여 화면을 새로 고친다.

f) r 누른 상태에서 이동

확인 방법 :

* r 키를 눌러 회전 모드를 활성화한다.
* 빨간색 중심점을 클릭한 상태에서 마우스를 움직인다.
* 다각형과 중심점이 함께 이동한다.

구현 방법 :

* keyboard 함수에서 r 키 입력 시 st.rotation\_mode를 토클한다.
* mousepress와 mousemove 함수에서 st.rotation\_mode 상태에서도 동일한 이동 로직을 적용한다.

g) 마우스 방향 따라 확대/축소

확인 방법 :

* 마우스 휠을 위로 돌리면 다각형이 확대된다.
* 마우스 휠을 아래로 돌리면 다각형이 축소된다.

구현 방법 :

* mouseWheel 함수에서 마우스 휠 방향(direction)을 확인하고, 확대/축소 비율(SCALE\_FACTOR)을 적용한다.
* scaling\_polygon 함수를 사용하여 다각형의 모든 점을 무게중심을 기준으로 스케일링한다.
* glutPostRedisplay를 호출하여 화면을 새로 고친다.

h) r 누른 상태에서 크기 확대/축소

확인 방법 :

* r 키를 눌러 회전 모드를 활성화한다.
* 마우스 휠을 돌려 다각형의 크기를 조정한다.

구현 방법 :

* mouseWheel 함수에서 st.rotation\_mode 상태에서도 확대/축소 로직을 동일하게 적용한다.
* scaling\_polygon 함수를 사용하여 다각형의 모든 점을 무게중심을 기준으로 스케일링한다.
* glutPostRedisplay를 호출하여 화면을 새로 고친다.