**과목명: 기초 컴퓨터 그래픽스**

**1분반**

**<<Homework #3>>**

**서강대학교 [컴퓨터공학과]**

**[20171666]**

**[이예은]**

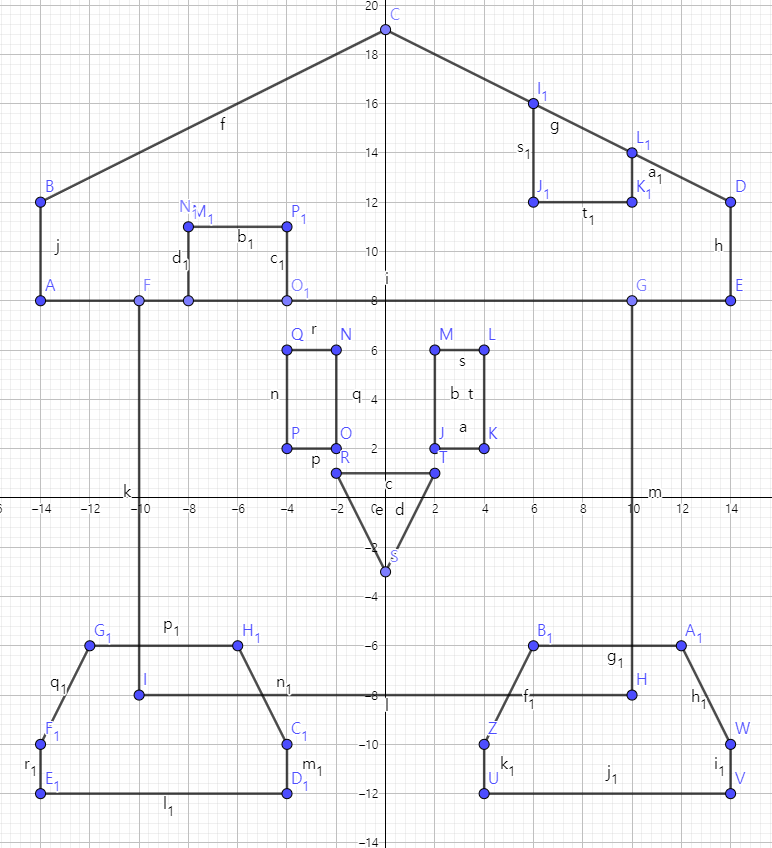
1. 기본적인 게임 방법

방향키를 움직이면 이에 따라서 모자가 움직입니다. 스페이스바를 누르면 칼이 날라가고 칼이 다른 물체들에 맞으면 그 물체들이 터집니다. 단, 집에 맞으면 칼이 튕길 수 있으니 주의해서 플레이 합니다.

1. 기본 5개 물체에 대한 동적 효과 구현
2. Hat: 유저의 입력에 따라 움직입니다. 방향키의 방향에 따라서 모자를 rotate 하고 이전 방향과 같은 방향의 입력이 들어오면 그 방향으로 이동합니다. 스페이스 바를 누르면 일정 시간동안 반시계방향으로 5도씩 rotate되어 총 30도 회전했다 다시 시계방향으로 5도씩 rotate되며 원상태로 돌아옵니다. 이를 통해 모자 속에서 칼이 나오는 듯한 애니메이션을 구현했습니다.
3. Sword: 스페이스 바를 누르면 hat의 위치에서 hat의 방향으로 직선 이동합니다. 아무 물체도 만나지 않았다면 화면 끝까지 이동했다 초기 위치로 돌아옵니다. 물체를 만났다면, 그 자리에서 칼은 사라지고 다시 초기 위치로 돌아갑니다. 하지만 만약 집을 만났다면 튕긴 것을 표현하기 위해 만난 자리에서 일정 시간동안 제자리에서 10도씩 회전 후 사라집니다.
4. House: 크기가 시간에 따라 커지다 6의 크기가 되면 작아지고, 2의 크기가 되면 다시 커지는 scale 운동을 반복합니다. 칼의 방해요소로 사용하기 위해 한 자리에서 계속 크기 변환을 반복하는 것이 아닌 랜덤으로 좌표를 이동합니다. 2개의 집은 서로 다른 시간 주기를 가지고 랜덤으로 좌표를 이동하며 크기 변환을 계속합니다.
5. Airplane: 기본적으로 대각선 지그재그 방향으로 이동합니다. 일정시간 동안 (x, y) = (2,1)의 비율로 움직이고 다음 일정 시간동안 (x, y) = (1,2)의 비율로 움직입니다. 지그재그 방향 이동시 (2,1)의 비율이라면 120도, (1,2)의 비율이라면 150도 물체를 회전시켜 이동 방향에 맞게 물체를 rotate 해줍니다. 또한 일정 시간 주기에 따라 중간에 360도 회전변환 하며 이동합니다. 칼을 만났다면 사라진 후 초기 위치로 이동해 다시 반복합니다.
6. Car: 아래 방향으로 sin 함수에 따라 이동 운동을 합니다. y좌표는 시간에 따라 1.2f씩 증가하며 그에 맞춰 sin값을 계산해 x에 넣어 sin 함수의 움직임을 보이도록 합니다. 또한 차가 달려온다는 것을 시각적으로 표현하기 위해 시간에 따라 점점 아래로 다가올수록 0.01f씩 scale 변환을 하여 점점 크기가 커지도록 구현하였습니다. 칼을 만났다면 사라진 후 초기 위치로 이동해 다시 반복합니다.
7. 새로운 물체 모델링 및 이에 대한 동적 효과 구현

Mushroom: 버섯 모양을 모델링 하였습니다. 버섯의 모자, 몸, 눈, 입, 발, 무늬를 모델링 하였고 이를 이용해 동적 효과를 구현하였습니다.

버섯 좌표 모델링



버섯은 기본적으로 sin 함수에 따라 곡선으로 통통 튀는 듯한 운동을 합니다. 작게 뛰었다 크게 뛰는 운동을 반복하며 x는 매시간 -1.2f만큼 움직이고 이에 따라 sin 값을 계산하여 y에 넣어줍니다. 이 때, 통통 튀는 움직임을 구현하기 위해 sin 값이 -가 되면 -로 계산하지 않고 +로 바꾸어 계산해줍니다. 화면의 오른쪽 끝에 도달했을 경우 초기 위치가 아닌 y좌표가 아래로 +200만큼 이동한 곳에서 시작합니다. x좌표는 초기 위치와 같습니다. 이동 후 튀는 운동을 반복합니다. 화면 아래쪽 끝에도 도달했을 경우에 초기 위치로 돌아갑니다. 버섯은 일정 시간 주기에 따라 중력이 바뀐 듯한 움직임을 보여줍니다. 일정 시간 주기에 따라 일정 시간동안 5도씩 rotate하여 총 180도만큼 rotate하고 이 방향에 맞추어 통통 튀는 움직임을 보이도록 180도 회전된 상태에서는 y값을 반대로 넣어줍니다. 칼을 만났다면 사라진 후 초기 위치로 이동해 다시 반복합니다.

Etc. fragment: 칼과 물체가 만났을 시 생기는 파편입니다. 칼과 물체가 만난 후 만난 지점에서 5개의 파편이 각자의 비율에 맞춰 일정 시간동안 위쪽 방향으로 이동하고, 일정 시간 이후 각자의 비율에 따라 아래로 퍼지는 움직임을 구현하였습니다. 파편 또한 하나라도 화면 밖을 벗어나면 사라지게 구현했습니다.