7주차 결과 보고서

전공: 컴퓨터공학 학년: 2학년 학번: 20221559 이름: 박준우

1. 코드를 템플릿으로 전부 제공받았기 때문에, 해당 코드의 흐름을 기준으로 작성하였다.

물은 주어진 선분을 따라 흐르거나, 선분 위에서 흐르지 않는다면 아래로 수직하게 흐르게 되고, 이는 선분들을 이은 모양이 될 것이다. 따라서 시작점부터, 이것들이 꺾이는 점들을 모두 찾아서 선분들을 그리면 된다.

따라서 꺾이는 점들을 Dot 구조체를 이용해 각 점의 좌표를 기록하고, 이를 기반으로 물의 시각적 흐름을 표현하게 된다.

이 경로 위의 점들이 각각 어디에 위치하게 될 지 구하는 메서드가 calculate\_path이다.

2. calculate\_path

첫 for-loop에서 start\_dot의 x좌표를 고정한 후, start\_dot의 y좌표를 1씩 늘려가며, 각각의 점에서 두번째 for-loop를 실행해, 각 선분들에 포함되어 있는지의 여부를 조사하게 된다. 따라서 위에서 먼저 만나는 선분이 있으면 해당 점에서 바로 실시하기 때문에, 아래에 있는 선분이 적합하다고 판단할 일은 없다.

현재 점에서 선분의 한 끝 점까지의 기울기와 선분의 기울기의 차이가 매우 작으면 (<EPSILON, 실수 연산의 오차 판별을 위해 필요함) 선분 위의 점이라고 판단할 수 있다. 해당 점을 경로에 저장하고, 이 선분의 기울기에 따라, 왼쪽 끝 점으로 흐를지, 오른쪽 끝 점으로 흐를지를 판단하여, 해당 점을 경로에 포함시킨 후, start\_dot으로 설정한다.

이렇게 한 선분이 끝난 후에는 다시 첫번째 for-loop에 따라 제어된다.

더이상 만나는 선분이 없고, 첫번째 for-loop의 update 구문에 따라 y좌표가 계속 커지던 중, 종료 조건에 걸리면, 해당 점도 경로에 포함시킨다.