Track Document

2020001658 이유민, 2021011158 김선희

code explanation for easy track

straight path

- if sensor4, sensor5 (middle sensor) == 1, go straight
- if sensor3 or sensor6 == 1 when sensor4, sensor5 are also == 1, modify to match the straight path by turning slightly

stop

- when sensor2, 3, 6, 7 and also sensor4,5 == 1, robot will stop at the moment

corner

- if sensor6, 7, 8 == 1, turn right by setting each wheel's direction to left_forward() and right_backward()
- if sensor1, 2, 3 == 1, turn left by setting each wheel's direction to left_backward() and right_forward()

else (out of the path)

- stop the robot by move(0,0)

code explanation for main track

corner, intersection : using sensorleft~ , sensorright~, allsensor~

- When the robot meets the intersection or corner, robot has priority in order of straight, left, and right.
- To check if that path is intersection and to decide the next step (go straight or turn left / right), go 2cm (which is default width of the path). If there is a path after that move, robot treats that as the sign of going straight according to the priority
- If the robot meets the case that it has to turn left or right, use turning method until the right_count or left_count reaches 160. Although the value corresponding to 90 degree rotation is 180 degrees, set to 160 to correct a slight error

winding road : using leftturn~, rightturn~, sensor1,2,3,6,7,8

- using forward / backward method to for smooth rotation stop

- Use stop_count to find the moment to stop the robot. We regard stoppoint~ sensor's locations as the starting line's length. When the stop_count will change to 2 after the robot completes two laps.

peer assessment

Name	Contribution
이유민	100
김선희	100