

# 로봇 네비게이션 기말프로젝트

조교 전형준

이번 기말프로젝트에서는 새로운 지도와 체크포인트, 임의의 장애물이 추가되었습니다. 지도는 그림 1과 같습니다.

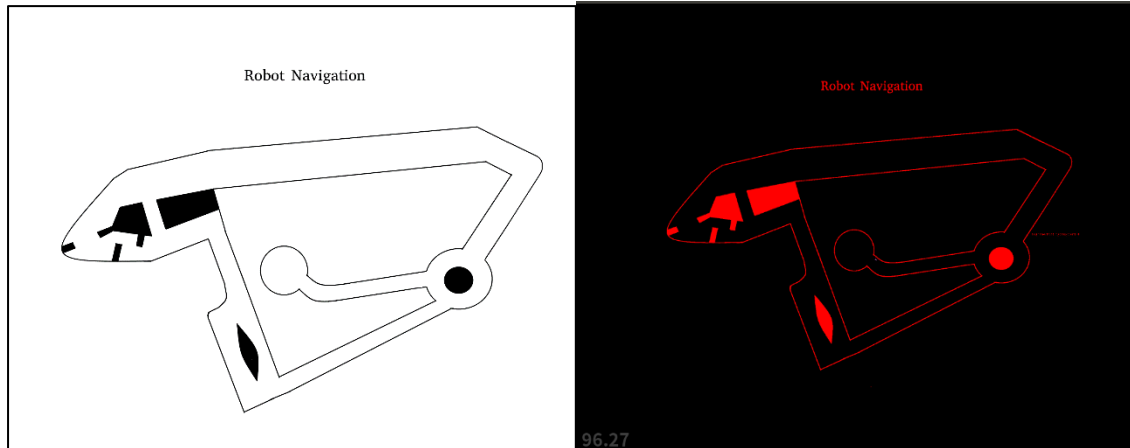


그림 1 지도와 시뮬레이션 환경

로봇은 그림 2의 초록색으로 표시된 지점에서 시작하여, 지도내 코스에서 빨간색 원으로 표시된 체크포인트들을 지나야 합니다. 저번과 마찬가지로 장애물과 충돌하거나 코스를 1회 완주할 경우에 코드가 종료되게 됩니다.

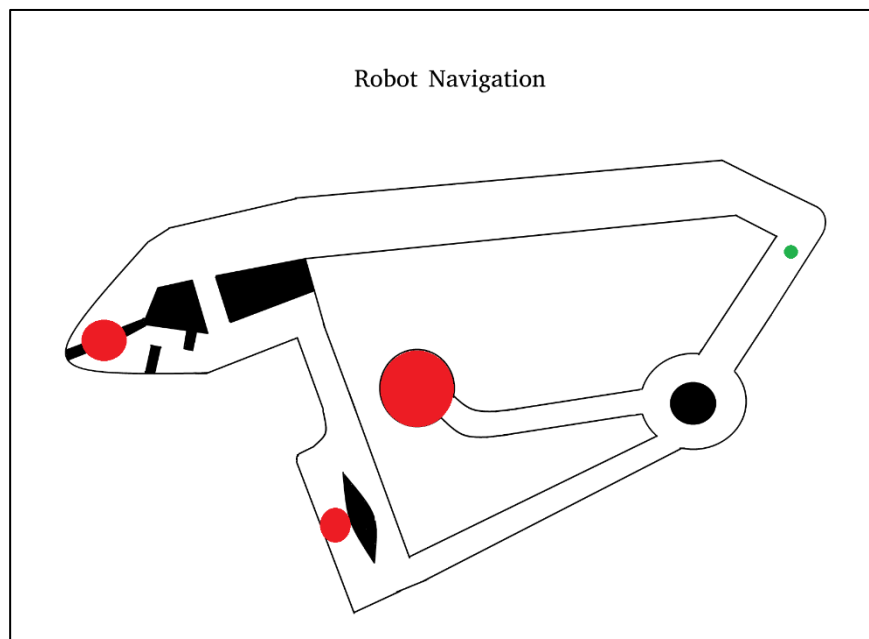


그림 2 출발점과 체크포인트

또한 학생분들에게는 제공되지 않지만, 그림 3의 파란색으로 표시된 영역에 추가적인 장애물을 넣을 예정입니다. 코스 주행 간에 미리 제공되지 않는 장애물도 피할 수 있는 라이다 기반의 주행 알고리즘을 사용하길 바랍니다. 예시는 오른쪽 이미지와 같습니다.

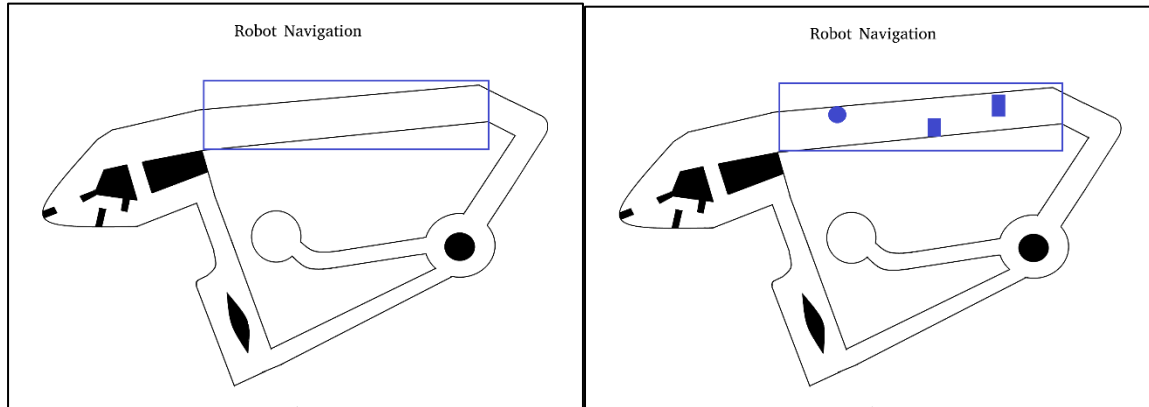


그림 3 임의의 장애물 영역 및 예시

## 주행 간에 사용가능한 정보

이번에 제공되는 정보는 저번에 사용했던 라이다 정보에 추가로, 지도와 지도상에 좌표를 제공할 예정입니다. 지도의 경우에는 이 안내서와 함께 첨부되어 있으며, png 형식의 이미지로 제공됩니다. 시뮬레이션 좌표는 지도 위의 좌표와는 다르기 때문에, 시뮬레이션상의 좌표는 그림 4를 참고하여 지도위의 좌표로 변환하시면 됩니다.  $(x_{sim}, y_{sim})$ 은 시뮬레이션상의 좌표( $x, y$ )이고,  $(x_{map}, y_{map})$ 은 지도상에 좌표( $x, y$ )입니다. **좌표축에 주의하시길 바랍니다.**

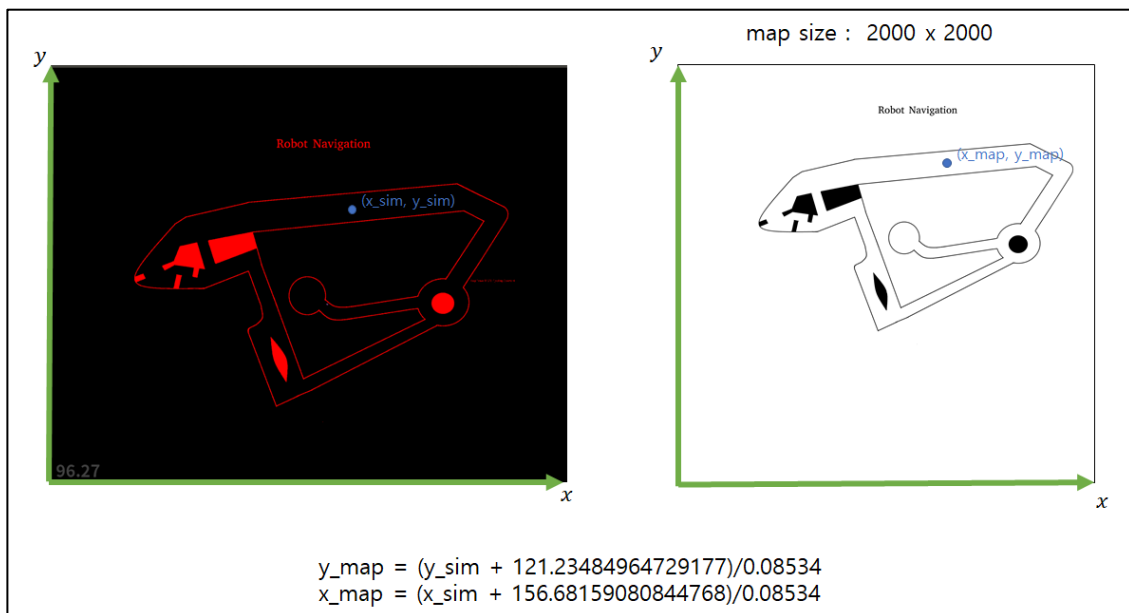


그림 4 좌표 변환 공식, 지도 좌표계 확인바람

시뮬레이션 상의 좌표는 기존의 process\_lidar 함수에 인자로 추가됩니다. 아래의 예시처럼 인자 coordinate = [x\_sim, y\_sim]를 통해서 로봇의 좌표를 얻으시면됩니다.

```
-----  
-  
class CustomDriver:  
    def process_lidar(self, ranges, coordinate):  
        x_sim = coordinate[0]  
        y_sim = coordinate[1]  
        #todo  
  
        speed = 5.0  
        steering_angle = 0.0  
        return speed, steering_angle
```

---

## 제출간 주의사항

- 코드 제출은 저번과 마찬가지로 drivers.py 파일과 추가적으로 설치해야 하는 패키지 requirement.txt에 작성하여 제출하시길 바랍니다.
- 사용하시는 지도는 'ROBOT\_NAVIGATION'임을 유의하시길 바랍니다.
- CustomDriver의 클래스로 코드 작성바랍니다.
- 제출 마감일은 12월 18일 23시 59분까지입니다.
- 점수 평가는 임의의 장애물을 추가한 지도내에 체크포인트를 따라서 장애물과 충돌없이 완주한 시간입니다. 시뮬레이션 상의 시간으로 채점합니다. (모두 같은 크기 및 위치의 장애물을 더할것입니다.)
- 제출은 아래의 이메일로 보내시길 바랍니다.
- [jeonkw@kw.ac.kr](mailto:jeonkw@kw.ac.kr)
- 보내신 코드에 대한 결과는 당일 다른 제출 코드와 함께 한번에 알려드리겠습니다.
- 자세한 채점기준은 채점 기준문서를 확인하시길 바랍니다.