

ROS_1일차_과제_3번_보고서_20기_인턴_2025402040_컴퓨터정보공학부_
노기문

1. 코드 분석(cpp,py공통)
2. 결과 화면

코드 분석

1. 핵심 통신 구조

Publisher:

/turtle1/cmd_vel 토픽에 geometry_msgs/msg/Twist 메시지를 발행.
→ 선속(linear.x)과 각속도(angular.z) 값을 주어 거북이 이동·회전을 제어.

Service Client:

/turtle1/set_pen 서비스에 turtlesim/srv/SetPen 요청.
→ RGB 색상, 선 굵기, 펜 on/off 설정.

2. 동작 방식

키 입력 처리

w/s/a/d → 전진, 후진, 제자리 좌회전, 제자리 우회전.
t/r/c → 삼각형, 사각형, 원 그리기.
q → 프로그램 종료.
=> 입력은 getch/getKey로 엔터 없이 즉시 한 글자씩 읽음.

move 함수

일정 시간 동안 선속·각속도를 주기적으로 퍼블리시.
예: (2.0, 0.0) → 직선 이동, (0.0, 1.57) → 제자리 회전.
이동이 끝나면 stop()으로 정지 명령(0 속도) 발행.

set_pen 함수

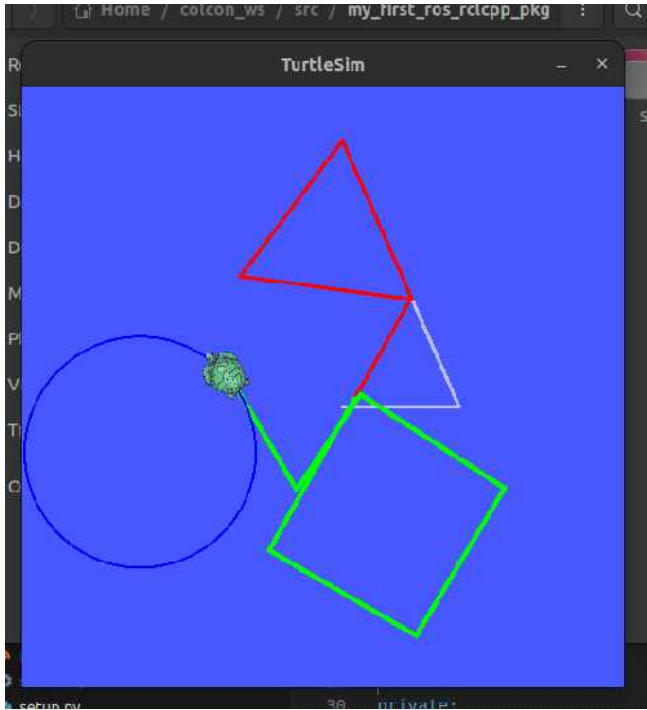
서비스 호출로 펜 색상과 굵기를 바꿈.
도형별로 색상·굵기 지정:
삼각형: 빨강, 굵기 3
사각형: 초록, 굵기 4
원: 파랑, 굵기 2

도형 그리기 함수

삼각형: 직선 이동 → 120도 회전 × 3번.
사각형: 직선 이동 → 90도 회전 × 4번.
원: 선속+각속도 동시에 주어 원 궤적을 그림.

결과 화면

1. hw3_result_cpp



2. hw3_result_py

