Project 3 by Tianhua Zhao

Gitlab: https://gitlab.com/tzhaojustdoit/student1819009

Motion primitives:

- Developed offline in c++
        - ■ Probably harder to do in c++ than in matlab or excel. I am not familiar with exporting to .txt file in matlab or excel.
- Bezier curve:
    - ○ Quintic bezier curve: (0, 0) (20, 0) (40, 0) (60, 3.7) (80, 3.7) (100, 3.7)
        - ■ By observation, these control points result in a smooth curve
- 3 motion primitives:
    - ○ Straight: displacement (100, 0) cost 100
    - ○ Left turn: displacement (100, 3.7) cost 105
    - ○ Right turn: displacement (100, -3.7) cost 105
        - ■ Having only 3 motion primitives results in better planning time. However the vehicle can only turn at certain x locations (multiple of 100m).
- 201 waypoints for each motion primitive

Graph configuration:

- Grid size : 20m * 3.7m
        - ■ Grid width is same as lane width. One grid column per lane makes the problem simple. Having multiple grid columns per lane would allow the vehicle to deviate from the center line, but it is not so necessary for this project.
- First grid's center is at (0,0) in map frame

Graph search algorithm:

- Weighted A* with epsilon = 1.1
        - ■ Not an incremental search since map is not changing.
        - ■ Better planning time than A*

Collision checking:

- Before planning, for each obstacle, compute grid coverage, mark occupied grids as blocked.
- Before planning, for each motion primitive, compute grid coverage relative to the start grid
- During planning, check if every grid covered by the motion primitive is free.
        - ■ Better planning time than checking every waypoint. Guarantees no collision.

Time used:

Motion primitive: 7 hours, collision checking 5 hours, obstacles, 5 hours, planner -getSuccessors 3 hours, planner-algorithm implementation 3 hours, map processing 5 hours

Total: around 30 hours.