

## Test Report (Planning)

### 1. Introduction of new interfaces and classes

Interfaces iMap and class Map: initialize the map, set start and end vertex, and compute the shortest path and its length.

Interfaces iDrive and class Drive: process perception response and control car's movement

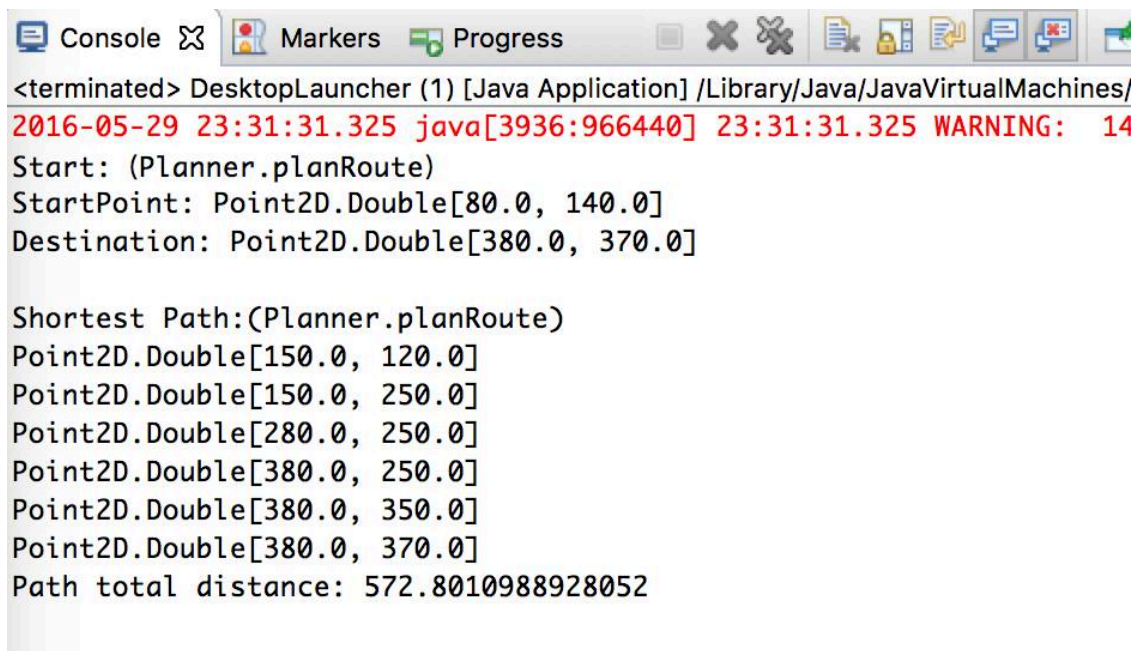
AI controller is modified by us for setting destination.

### 2. Self-testing for planning part

Before testing the integrated system, we tested our group's individual part: planning. The testing environment: Mac OS X El Capitan 10.11.4, IDE: Spring Tool Suite, jdk:1.8.0.

In our individual test, we did not have the perception results, so that we only tested part functionalities. To be specific, in terms of class Map's functionalities, the map can be initialized successfully, the start vertex and end vertex can be set correctly, the shortest path can be computed correctly, the length of path can be computed correctly. Also, in terms of class Drive's functionalities, the car can arrive the destination from the start automatically and correctly, the car can drive straightforward on the left lane, and turn into the left lane as well. Moreover, the estimated time of arrival can be computed correctly.

Test Sample Example: Start (80,140) Destination (380,370), test results are shown in the following screen shot.

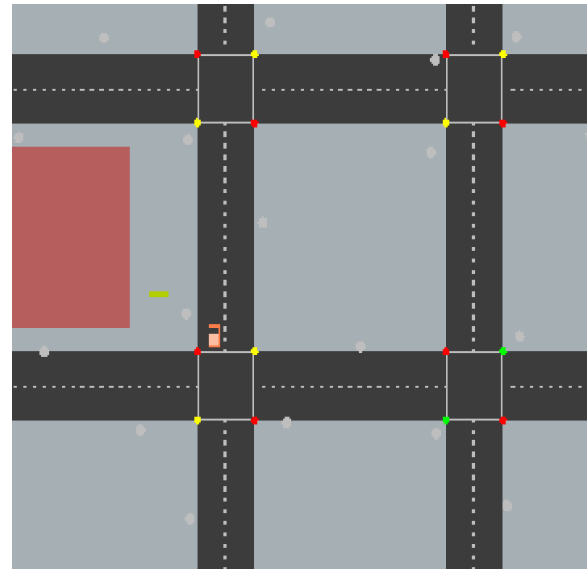
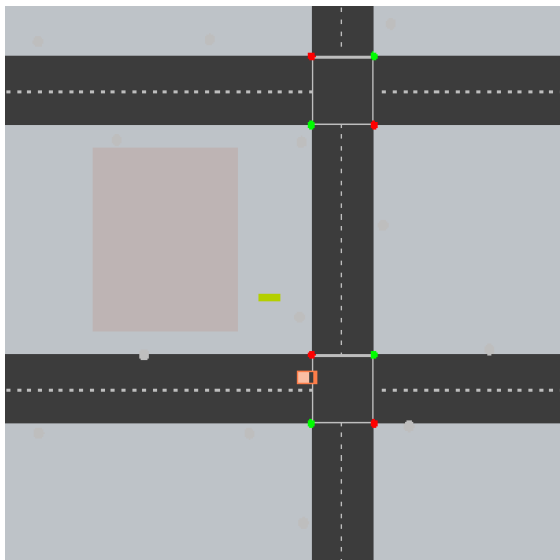


```
<terminated> DesktopLauncher (1) [Java Application] /Library/Java/JavaVirtualMachines/
2016-05-29 23:31:31.325 java[3936:966440] 23:31:31.325 WARNING: 14
Start: (Planner.planRoute)
StartPoint: Point2D.Double[80.0, 140.0]
Destination: Point2D.Double[380.0, 370.0]

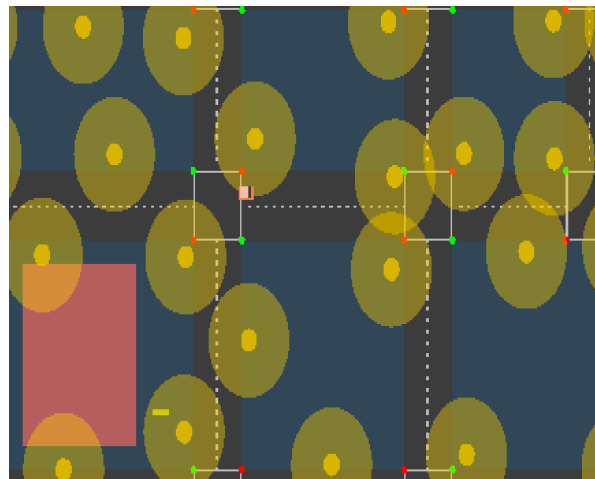
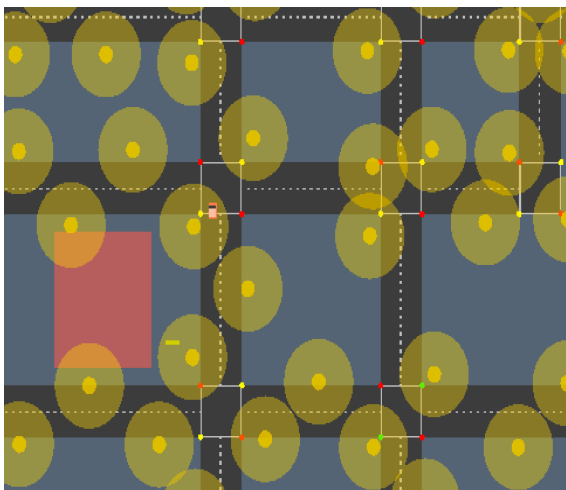
Shortest Path:(Planner.planRoute)
Point2D.Double[150.0, 120.0]
Point2D.Double[150.0, 250.0]
Point2D.Double[280.0, 250.0]
Point2D.Double[380.0, 250.0]
Point2D.Double[380.0, 350.0]
Point2D.Double[380.0, 370.0]
Path total distance: 572.8010988928052
```

Car's turning movements in each turning point are captured by screenshots, and they are listed in the following.

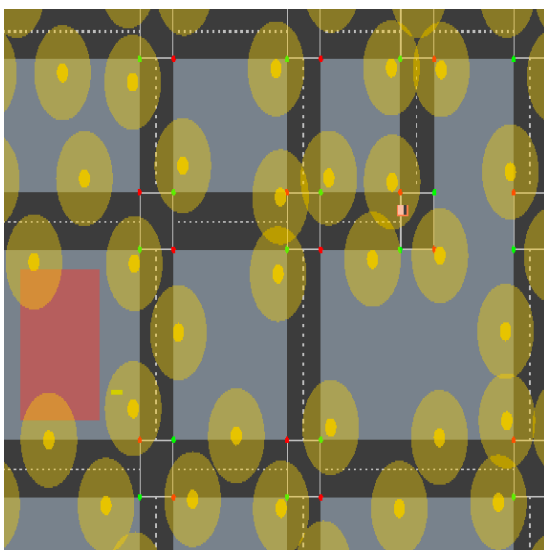
1. Passing the first turning point (Turn left to the left lane)



2. Passing the second turning point (Turn right to the left Lane)



3: Passing the third turning point (Turn left to the left Lane)



### **3. Integration system testing**

In the integration system test, we integrated sensing (from group 39) and perception (from group 41) functionalities with our planning functionality.

First of all, the AI Controller class file was modified based on our group's implementation. Thus our team unified the inconsistency of AI controller in order to set an available testing platform. After that, the testing platform was set up successfully and the whole system could run, but there was an important issue:

Perception group failed to pass the perception results of the traffic lights, so that we cannot process the first traffic object, which disables car's further normal operations.