**Problem 1 (Extra Credit) follows…**  
**Description:** Problem 1 is designed to demonstrate a complex Garmin by finding the minimum distance, best time and toll avoidance routes between a city and all other cities in the database. This includes reachable and unreachable vertices, edges and simple paths.

**Data File (California.in)**

6

Hartford Middletown Hamden

Waterbury Bristol Manchester

Hartford 3 Middletown 17 60 0 Bristol 19 55 1 Manchester 10 45 0

Middletown 3 Hartford 17 60 0 Hamden 21 55 1 Waterbury 24 65 0

Hamden 1 Middletown 21 55 1

Waterbury 2 Bristol 12 65 1 Middletown 24 65 0

Bristol 2 Waterbury 12 65 1 Hartford 19 55 1

Manchester 1 Hartford 10 45 0

**Program Output:**  
From Middletown (mileage, using minimum distance) to...

Hartford is 17 (Middletown->Hartford)

Middletown is 0 (Middletown)

Hamden is 21 (Middletown->Hamden)

Waterbury is 24 (Middletown->Waterbury)

Bristol is 36 (Middletown->Hartford->Bristol)

Manchester is 27 (Middletown->Hartford->Manchester)

From Waterbury (mileage, using minimum distance) to...

Hartford is 31 (Waterbury->Bristol->Hartford)

Middletown is 24 (Waterbury->Middletown)

Hamden is 45 (Waterbury->Middletown->Hamden)

Waterbury is 0 (Waterbury)

Bristol is 12 (Waterbury->Bristol)

Manchester is 41 (Waterbury->Bristol->Hartford->Manchester)

From Middletown (time, using minimum time) to...

Hartford is 17 (Middletown->Hartford)

Middletown is 0 (Middletown)

Hamden is 22 (Middletown->Hamden)

Waterbury is 22 (Middletown->Waterbury)

Bristol is 33 (Middletown->Waterbury->Bristol)

Manchester is 30 (Middletown->Hartford->Manchester)

From Waterbury (time, using minimum time) to...

Hartford is 31 (Waterbury->Bristol->Hartford)

Middletown is 22 (Waterbury->Middletown)

Hamden is 44 (Waterbury->Middletown->Hamden)

Waterbury is 0 (Waterbury)

Bristol is 11 (Waterbury->Bristol)

Manchester is 44 (Waterbury->Bristol->Hartford->Manchester)

From Middletown (tolls, avoiding tolls) to...

Hartford is 17 (Middletown->Hartford)

Middletown is 0 (Middletown)

Hamden is 9999 (Middletown->Hamden)

Waterbury is 24 (Middletown->Waterbury)

Bristol is 10016 (Middletown->Hartford->Bristol)

Manchester is 27 (Middletown->Hartford->Manchester)

From Waterbury (tolls, avoiding tolls) to...

Hartford is 41 (Waterbury->Middletown->Hartford)

Middletown is 24 (Waterbury->Middletown)

Hamden is 10023 (Waterbury->Middletown->Hamden)

Waterbury is 0 (Waterbury)

Bristol is 9999 (Waterbury->Bristol)

Manchester is 51 (Waterbury->Middletown->Hartford->Manchester)

**Answers to Questions:**

1. Among multiple choices of a path, your program chooses the minimum distance, fastest time and toll road avoidance.

Specifying the start location Middletown…

Minimum distance is used: Hartford is 17 (Middletown->Hartford) because Middletown has more than just one path to get to Hartford. It could have taken a longer distance from Middletown->Waterbury->Bristol->Hartford but it chose the minimum distance instead.

Fastest time is used: Bristol is 33 (Middletown->Waterbury->Bristol) because Middletown has more than just one path to get to Bristol. It could have taken a longer time from Middletown->Hartford->Bristol but it chose the fastest time instead.

Toll road avoidance is used: Hartford is 17 (Middletown->Hartford) because Middletown has more than just one path to get to Hartford. It could have taken a route with tolls from Middletown->Waterbury->Bristol->Hartford but it chose the route without tolls instead.

Specifying the start location Waterbury…

Minimum distance is used: Hartford is 31 (Waterbury->Bristol->Hartford) because Waterbury has more than just one path to get to Hartford. It could have taken a longer distance from Waterbury->Middletown->Hartford but it chose the minimum distance instead.

Fastest time is used: Bristol is 11 (Waterbury->Bristol) because Waterbury has more than just one path to get to Bristol. It could have taken a longer time from Waterbury->Middletown->Hartford->Bristol but it chose the fastest time instead.

Toll road avoidance is used: Hartford is 41 (Waterbury->Middletown->Hartford) because Waterbury has more than just one path to get to Hartford. It could have taken a route with tolls from Waterbury->Bristol->Hartford but it chose the route without tolls instead.

1. Your one-way path(s) are taken only when they should be and avoided when

they should not have been taken.

Specifying the start location Middletown…

One-way path is taken only because it provides minimum distance route: Hartford is 17 (Middletown->Hartford)

Specifying the start location Waterbury…

One-way path is taken only because it provides minimum distance route: Bristol is 11 (Waterbury->Bristol)

While there is no best case example for avoiding one-way paths in my data set, its effect can still be seen.

Specifying the start location Middletown…

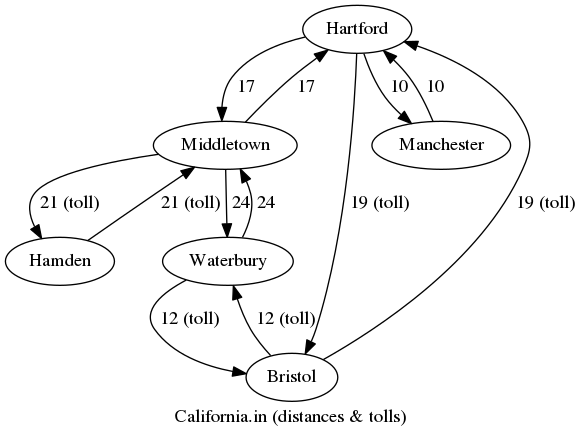
This one-way path is the only option because there is no other path from Middletown to Hamden, certainly not one that does not involve any tolls, but toll avoidance is proven to work in question 5(a) above: Hamden is 9999 Middletown->Hamden)

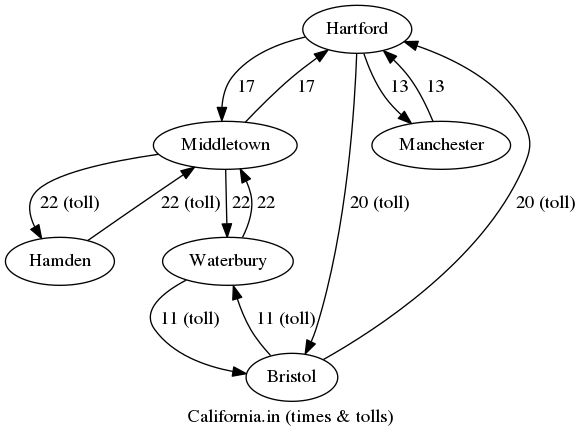
Specifying the start location Waterbury…

This one-way path is the only option because there is no other path from Waterbury to Bristol, certainly not one that does not involve any tolls, but toll avoidance is proven to work in question 5(a) above: Bristol is 9999 (Waterbury->Bristol)

If my data set had a one-way path with a toll and an alternative path with multiple stops all without tolls, then the one-way path would be avoided and path with multiple stops taken.

**Complex Garmin:**

****



**Problem 2 follows…**  
**Description:** Problem 2 is designed to demonstrate a Bellman-Ford by finding the route between two cities in a database (LA and Boston) with the least cost. This includes changing “cost” by increments until the route changes from a direct flight to one which includes multiple stops.

**Data File (TravelToBoston.in):**

6

LA Chicago Boston Dallas CarSalesman Waiter

LA 3 Chicago 320 Dallas 500 Boston 700

Chicago 3 Boston 400 Dallas 300 CarSalesman -125

CarSalesman 2 Boston 500 Dallas 280

Dallas 2 Boston 450 Waiter -275

Waiter 1 Boston 490

Boston 0

LA Boston

Waiter Dallas

Dallas

**Program Output:**  
LA: Boston 700 Chicago 320 Dallas 500

Chicago: Boston 400 CarSalesman -125 Dallas 300

Boston:

Dallas: Boston 450 Waiter -275

CarSalesman: Boston 500 Dallas 280

Waiter: Boston 490

Your choices for a source vertex are:

1 - LA

2 - Chicago

3 - Boston

4 - Dallas

5 - CarSalesman

6 - Waiter

Enter a number from 1 to 6: 1

Your choices for a destination vertex are:

0 - All vertices

1 - LA

2 - Chicago

3 - Boston

4 - Dallas

5 - CarSalesman

6 - Waiter

Enter a number from 1 to 6: 3

Cost to Boston: 690, Path starting in LA

-> Chicago 320

-> CarSalesman 195

-> Dallas 475

-> Waiter 200

-> Boston 690

**Bellman-Ford:**

