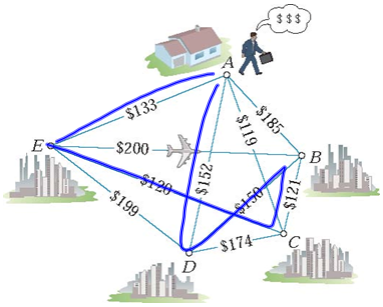
Strategy 1 (Exhaustive Search)

the **brute-force algorithm**



Determine cost of every possible route.
Pick cheapest.

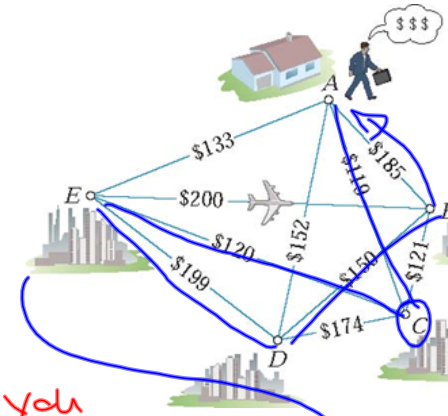
Route
A-ECBD-A add up the cost ....

4! choices for the order of other cities.
24 possible routes
pick cheapest.

with 19 cities to visit
really big 1.2164 E 17
Home city and 19 others 121,640,000,000,000,000

Strategy 2 (Go Cheap)

the **nearest-neighbor algorithm**



1. Start at home in city A
2. Select cheapest flight from where you are to one of the destinations unvisited
   if all cities visited, then go home.
3. Take that flight (follow that route)
4. Go to step 2.

A-C 119
C-E 120
E-D 199
D-B 150
B-A 185
Total

Algorithm 1: The Brute-Force Algorithm

- Step 1.
- Step 2.
- Step 3.

Algorithm 2: The Nearest-Neighbor Algorithm

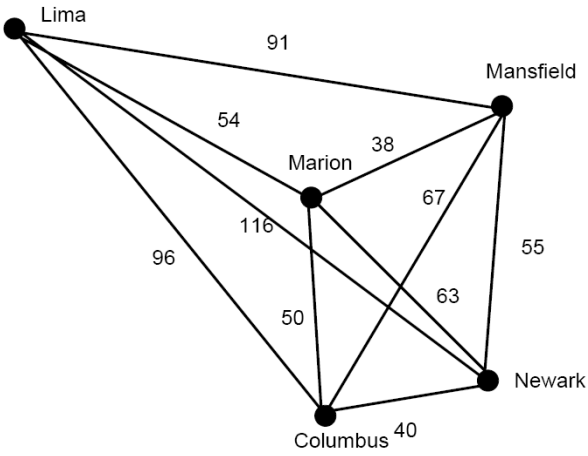
Start.

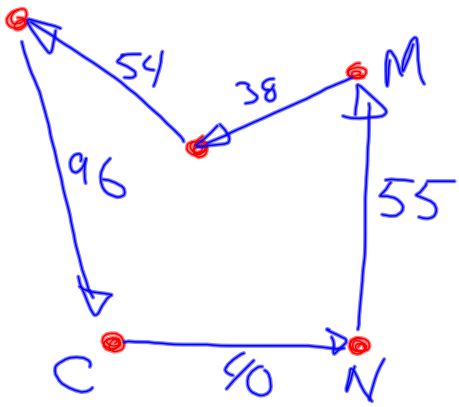
First step.

Middle steps.
Repeat this until all the vertices have been visited. Then take last edge back to starting vertex.

Suppose Ohio State's President Gee wants to visit all of the campuses: Columbus, Lima, Marion, Mansfield, a Newark. He'd like to make one tour, and keeping his driving mileage to a minimum.

Find a Hamilton Circuit with Columbus as the start and finish using Nearest Neighbor.

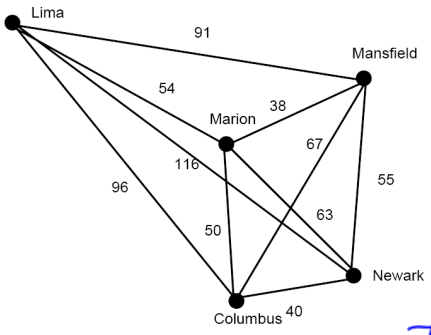


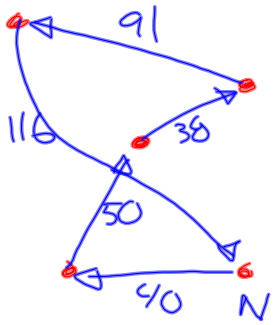


Is this route "optimal"?

Total 283 miles.

Do the same thing except suppose you start and end in Newark. What happens?





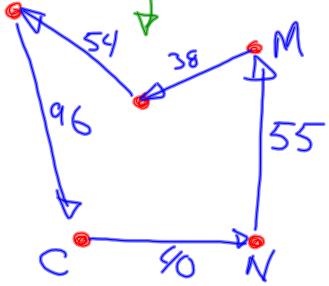
Total Distance = 335

Nearest Neighbor Doesn't always work

Easy, Quick Algorithm. but not always the best.

If we have to start in Newark is this best possible)

No, this is shorter.



The brute-force algorithm is an **Inefficient algorithm:**

List all possible routes.
Pick cheapest one

How many steps for a 5-city problem? $4! = 24$ routes to check

How many for a 10-city problem? $9! = 362,880$

How many for a 50-city problem? $49! = 6.08 \times 10^{62}$

Too Big

The nearest-neighbor algorithm is an **efficient algorithm**.

How many edges do you have to check at each step in a 5-city problem (at most)? 4 cities
How many steps are there?
How many total edge-checks are there?

What about a 50-city problem? 50 cities at most

5 cities each with at most 4 edges to check

at most 20 things to check.

49 edges to check.

2450 total things to check at worst.

Much more efficient than Brute Force.

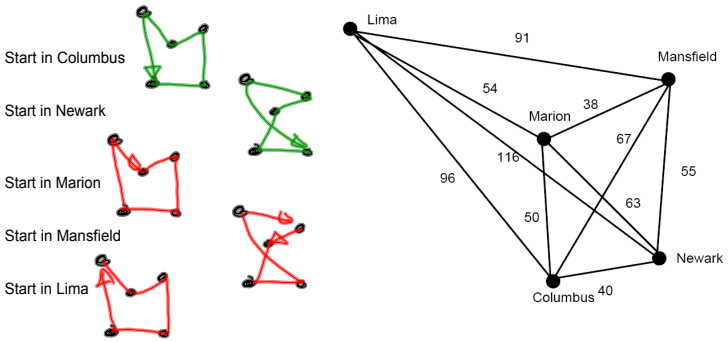
But.... Not always optimal

A really good algorithm for solving TSP's in general would have to be both efficient (like the nearest-neighbor) and *optimal* (like the brute-force). Unfortunately, nobody knows of such an algorithm.

We will use the term **approximate algorithm** to describe any algorithm that produces solutions that are, most of the time, reasonably close to the optimal solution.

Algorithm 3: The Repetitive Nearest-Neighbor Algorithm

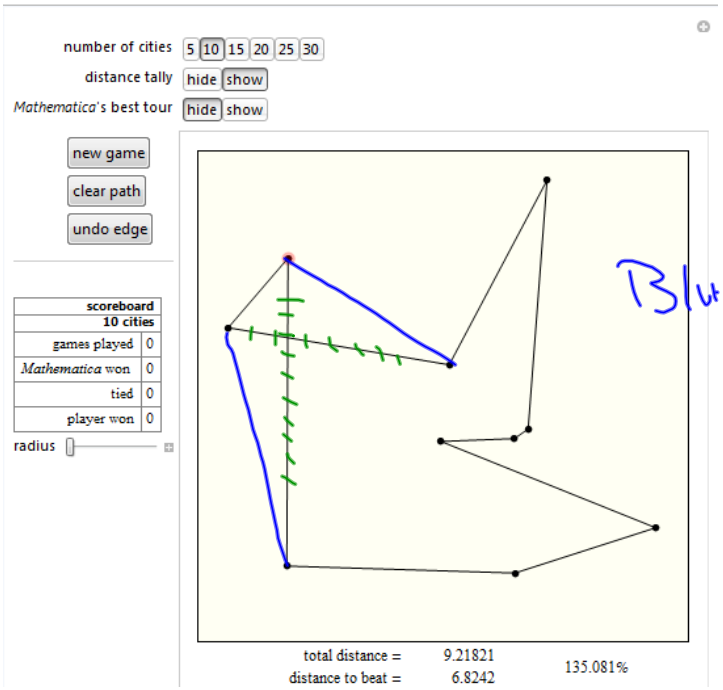
My Home city is Marion.



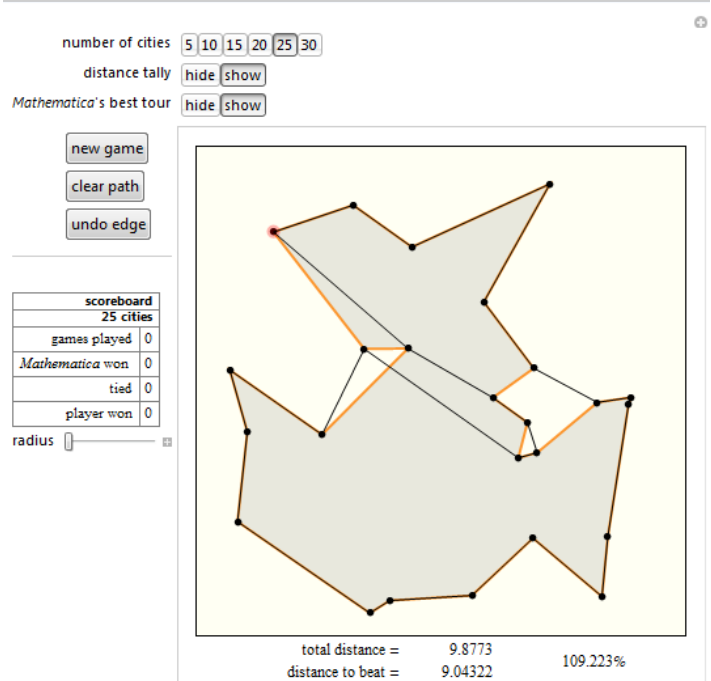
Algorithm: Repeat Nearest Neighbor each time starting at different city.
Get a list of reasonable routes
Pick best one of those.

Fast and efficient, but won't always get optimal solution.

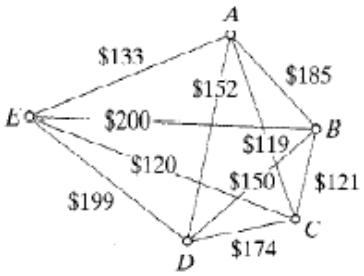
(Better than single Nearest Neighbor)

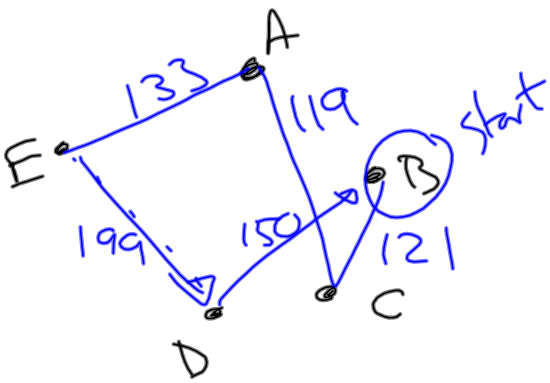


Blue route better than green.



1. For the weighted graph shown in the figure, (i) find the indicated circuit, and (ii) give its cost. (This is the graph discussed in Example 6.7.)



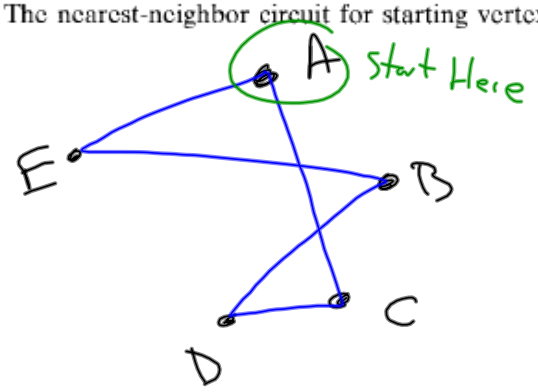


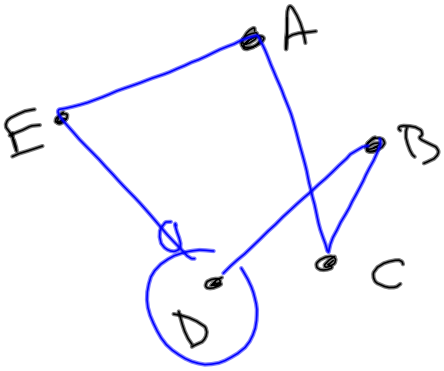
(a) The nearest-neighbor circuit for starting vertex B

(b) The nearest-neighbor circuit for starting vertex C

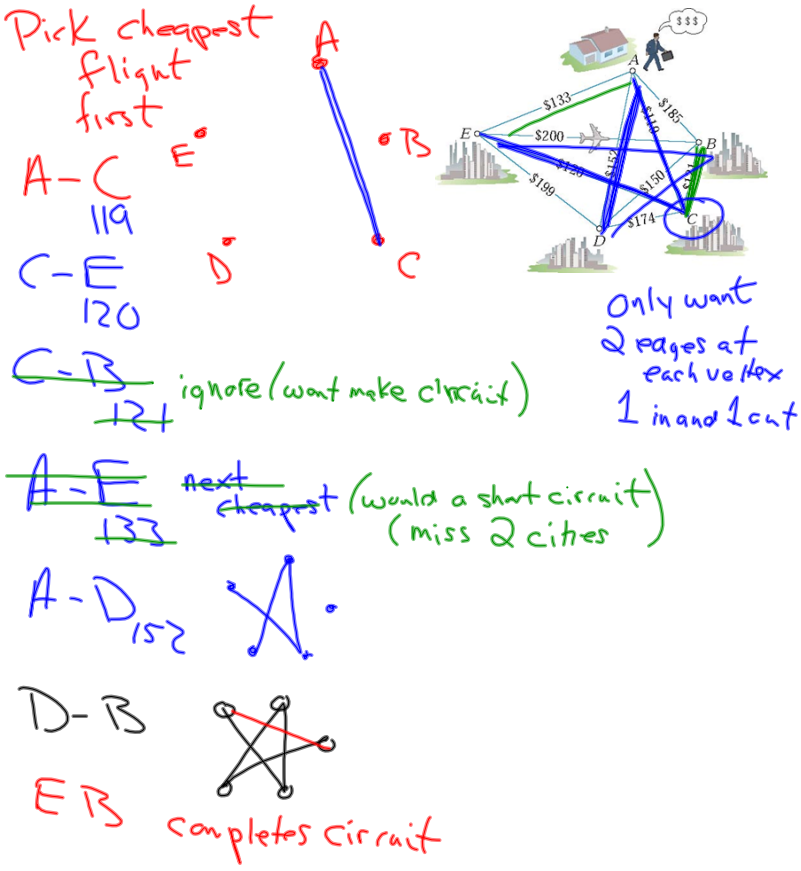
(c) The nearest-neighbor circuit for starting vertex D

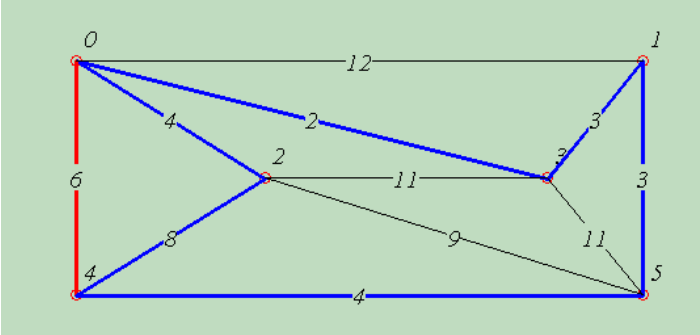
(d) The nearest-neighbor circuit for starting vertex E

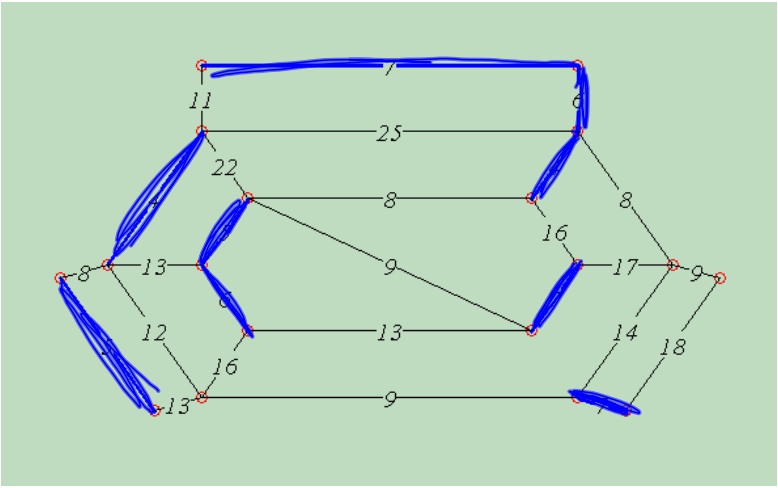


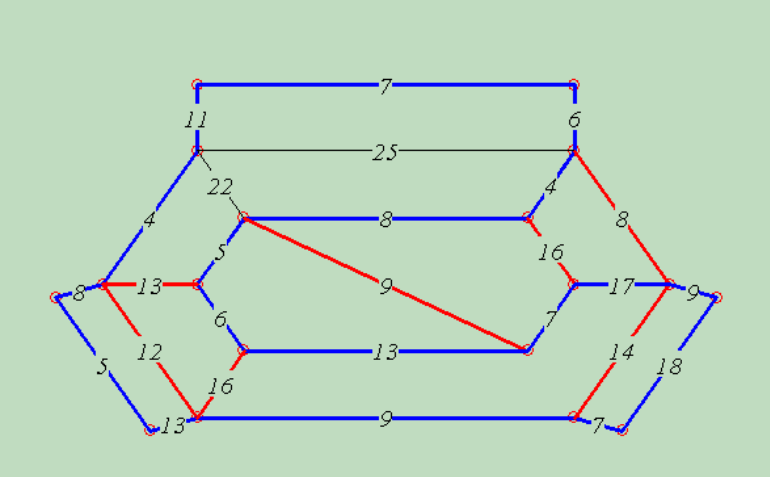


Algorithm 4: The Cheapest-Link Algorithm



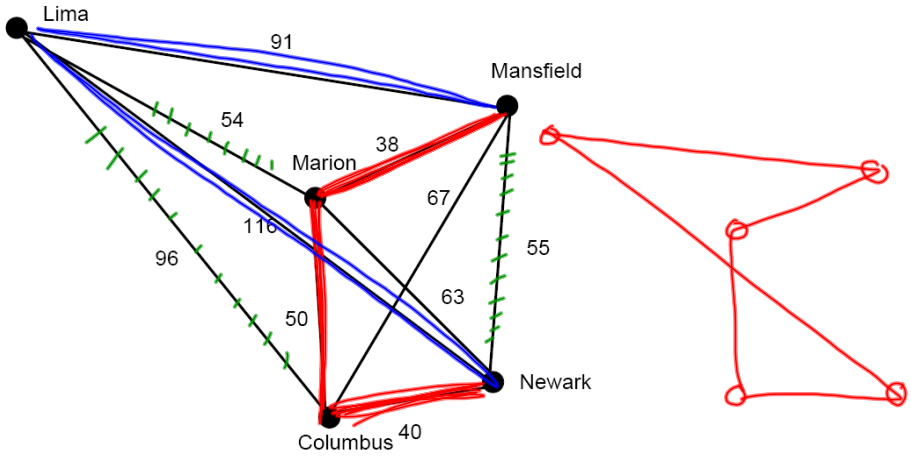

0
1
12
4
2
3
2
3
6
11
3
8
9
11
4
5
4


7
11
6
25
22
8
8
16
8
13
9
17
9
6
12
13
14
18
16
13
9



Suppose Ohio State's President Holbrook wants to visit all of the campuses: Columbus, Lima, Marion, Mansfield, and Newark. She'd like to make one tour, and keeping her driving mileage to a minimum.

Find a Hamilton Circuit using Cheapest Link.



http://www-e.uni-magdeburg.de/mertens/TSP/TSP.html

Click on "Nearest Neighbor Heuristic"
Click "Run" or "Step" to do cheapest link
Then Reset and hit solve to show optimal circuit