

# Exercise/Quiz/Homework

---

- Combining today's quiz with a new homework assignment.
- We will discuss the problem
  - some approaches towards solving the problem.
  - why the problem is "hard".
  - what should we expect when we "solve the problem"

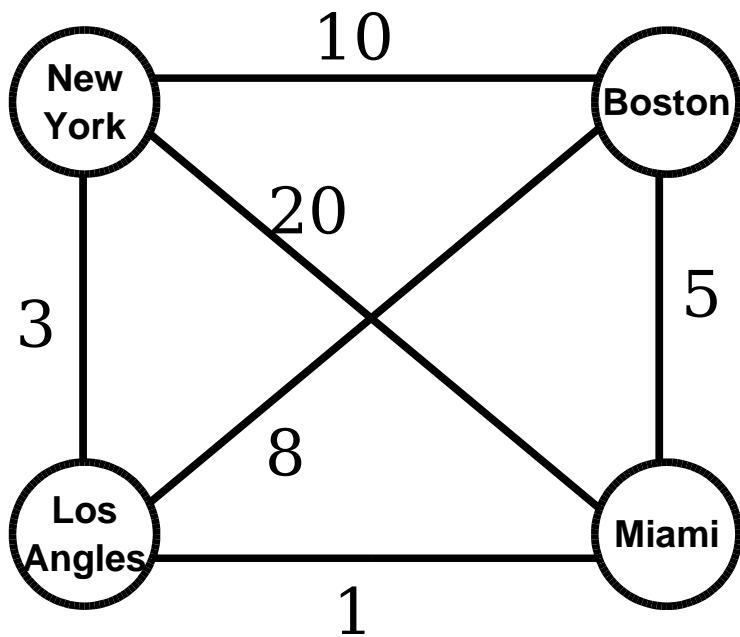
# Traveling Salesman Problem

---

- A salesman has a list of cities that he/she must visit.
- There is a "cost" to travel between cities.
  - there is a table of the costs available.
- Find the "cheapest" route that visits every city exactly once, and returns to the starting city.
  - any city can be the starting city.

# Example Costs

---



	New York	Boston	Miami	L.A.
New York	0	10	20	3
Boston	10	0	5	8
Miami	20	5	0	1
L.A.	3	8	1	0

# Some Issues

---

- How to represent the table of costs?
- How could we represent a "tour"?
- How can we compute the cost of a "tour"?
- We need to generate all possible tours
  - or do we?
- What difference does the starting city make?
-

# Data Structures

---

- Lists, structures, trees, list of structures, structures that contain lists that contain structures, etc...  
What is appropriate?
- Do we need to consider efficiency?
  - consider difference between two search algorithms we discussed – dramatic difference in "search time" when using a tree vs. a list.
- Data Definitions!

# Solving a TSP

---

- In general, we need to check out all possible tours.
- How many possible tours are there?
  - 4 cities?
  - 10 cities?
  - 100 cities?
- The number of tours we need to consider grows much faster than the number of cities.

# Exhaustive Search vs. Heuristics

---

- We will consider only exhaustive search
  - try all possible tours and return one that has the shortest cost (there may be many).
- In real life people often approach such problems by using some general rules to guide the search (the rules are called heuristics).
  - In most situations this approach is not guaranteed to lead to an optimal answer.
  - In many situations this is the only feasible approach.

# Assignments

---

1) Exercise & Quiz Today: develop data definitions for all the data structures you will need.

- submit them to WebCT.

2) Homework due in 1 week: complete the design and implementation.

- use test data provided to check your code.
- Submit complete code to WebCT.



# Test Data: 4 cities

## Minimum Cost: 19

---

City	0	1	2	3
0	0	10	20	3
1	10	0	5	8
2	20	5	0	1
3	3	8	1	0

# Test Data: 5 cities

## Minimum Cost: 45

---

City	0	1	2	3	4
0	0	10	11	10	9
1	10	0	11	9	12
2	11	11	0	11	11
3	10	9	11	0	5
4	9	12	11	5	0

# Test Data: 6 cities

## Minimum Cost: 28

---

City	0	1	2	3	4	5
0	0	3	10	5	6	10
1	3	0	5	3	11	6
2	10	5	0	8	5	5
3	5	3	8	0	7	6
4	6	11	5	7	0	12
5	10	6	5	6	12	0

# Test Data: 7 cities

## Minimum Cost: 35

---

City	0	1	2	3	4	5	6
0	0	4	9	3	6	9	11
1	4	0	10	12	7	12	11
2	9	10	0	7	6	4	12
3	3	12	7	0	10	6	4
4	6	7	6	10	0	7	12
5	9	12	4	6	7	0	7
6	11	11	12	4	12	7	0

# Test Data: 8 cities

## Minimum Cost: 36

---

City	0	1	2	3	4	5	6	7
0	0	7	5	8	5	10	5	3
1	7	0	10	12	9	12	3	8
2	5	10	0	5	8	5	6	12
3	8	12	5	0	12	8	4	8
4	5	9	8	12	0	7	8	4
5	10	12	5	8	7	0	8	3
6	5	3	6	4	8	8	0	4
7	3	8	12	8	4	3	4	0

Depending on how you implement the TSP search, this one could take a while (a few minutes).