

Topcoder SRM 658, Division 1, 300 Points  
“OddEvenTree”

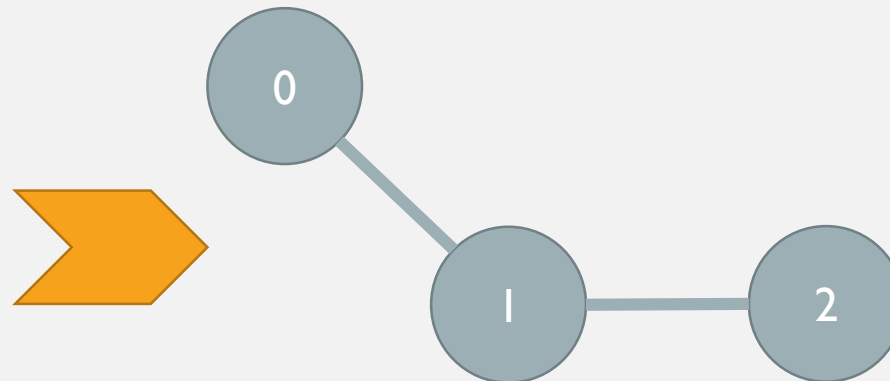
Elliot Greenlee  
EECS Department  
University of Tennessee

CS494 Presentation  
March 22, 2016

# Problem

- Is it possible to create a tree with the given properties?
  - Tree with  $N$  nodes and  $N-1$  edges
  - Distance is edges between two nodes
  - Distance is specified either odd or even

	0	1	2
0	Even	Odd	Even
1	Odd	Even	Odd
2	Even	Odd	Even



# Problem

	0	1	2	3	4
0	Even	Odd	Even	Odd	Even
1	Odd	Even	Odd	Even	Odd
2	Even	Odd	Even	Odd	Even
3	Odd	Even	Odd	Even	Odd
4	Even	Odd	Even	Odd	Even

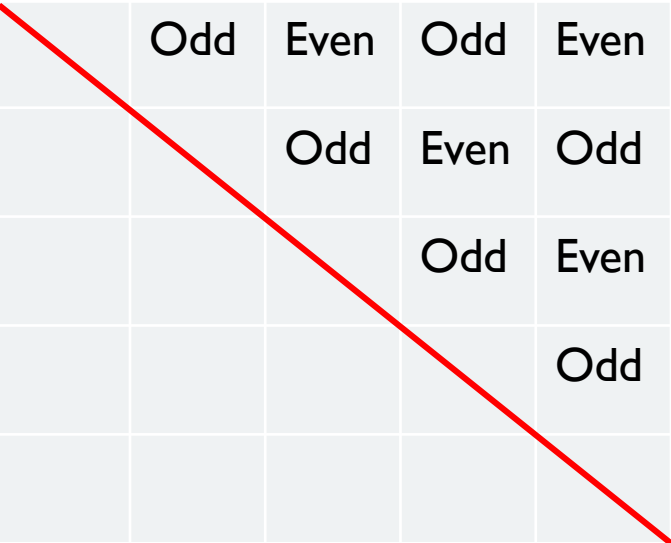
# Problem

	0	1	2	3	4
0		Odd	Even	Odd	Even
1	Odd		Odd	Even	Odd
2	Even	Odd		Odd	Even
3	Odd	Even	Odd		Odd
4	Even	Odd	Even	Odd	

Diagonal is always 0 nodes away and even

# Problem

	0	1	2	3	4
0		Odd	Even	Odd	Even
1			Odd	Even	Odd
2				Odd	Even
3					Odd
4					

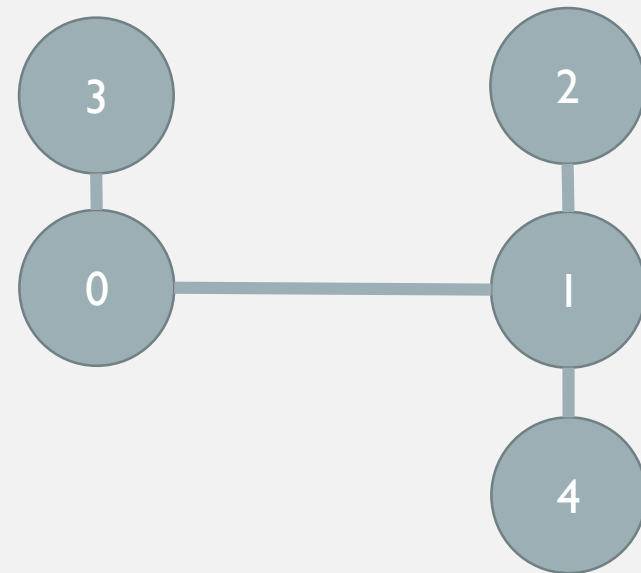


The reflection will be symmetrical

# Problem

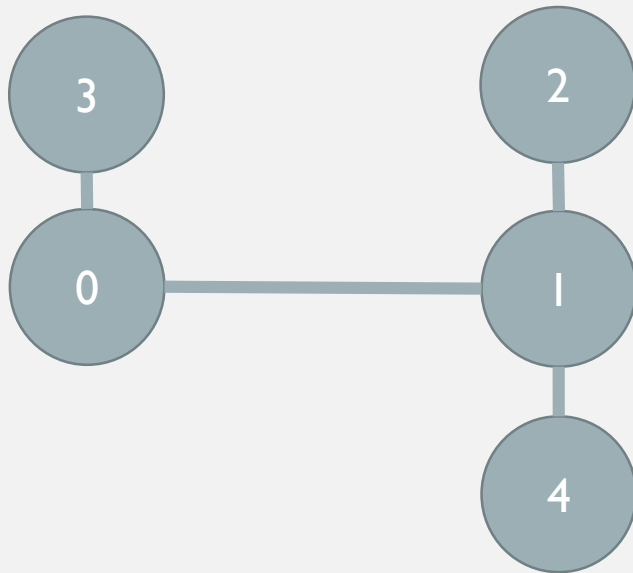
	0	1	2	3	4
0		Odd	Even	Odd	Even
1			Odd	Even	Odd
2				Odd	Even
3					Odd
4					

Create tree



# Problem

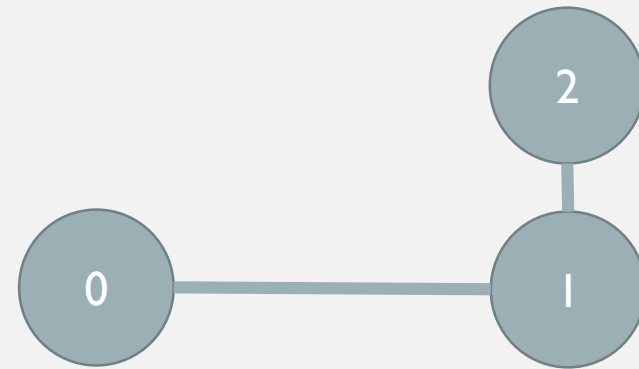
Return: (0, 3), (0, 1), (1, 2), (1, 4)



# Problem

	0	1	2
0		Odd	Even
1			Even

What happens when a graph is not valid?

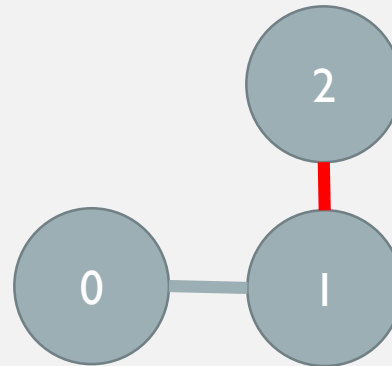




# Problem

	0	1	2
0		Odd	Even
1			Even

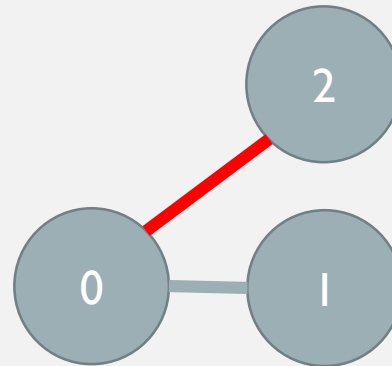
What happens when a graph is not valid?



# Problem

	0	1	2
0		Odd	Even
1			Even

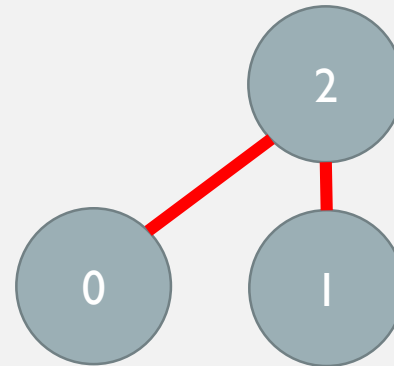
What happens when a graph is not valid?



# Problem

	0	1	2
0		Odd	Even
1			Even

What happens when a graph is not valid?



# Prototype and Variables

- Class name: OddEvenTree
- Method: getTree
- Parameters: vector <string>
- Return value: vector <int>
- Constraints:
  - N between 2 and 50, inclusive
  - x will contain N elements
  - Each string in x will have N characters
  - Each character in x will be 'O' or 'E'

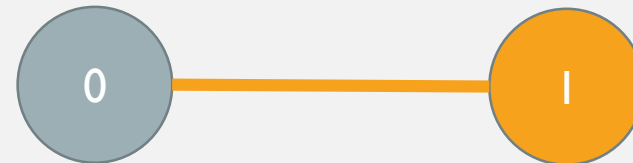
# Solution

- Insight 1: Simplifying the table
  - The diagonal information is removable
  - The reflected information is removable
- Insight 2: Odd/Even does not imply distance
  - N-1 distance limit

# Solution

- Insight 3: Start somewhere
  - Pick node 0 as the 'EvenNode'
  - One of node 0's odd nodes will be one away as the 'OddNode'

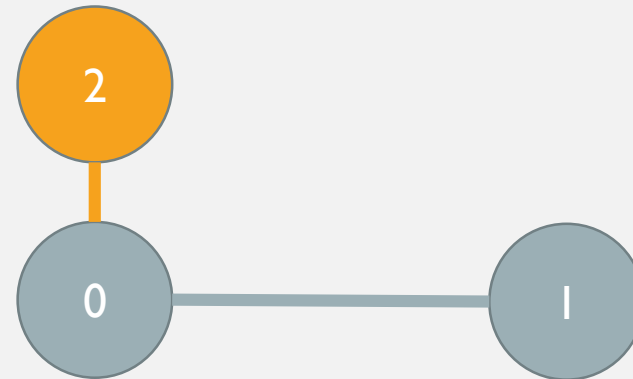
	0	1	2	3	4
0		Odd	Odd	Even	Even
1			Even	Odd	Odd
2				Odd	Odd
3					Even



# Solution

- Easy

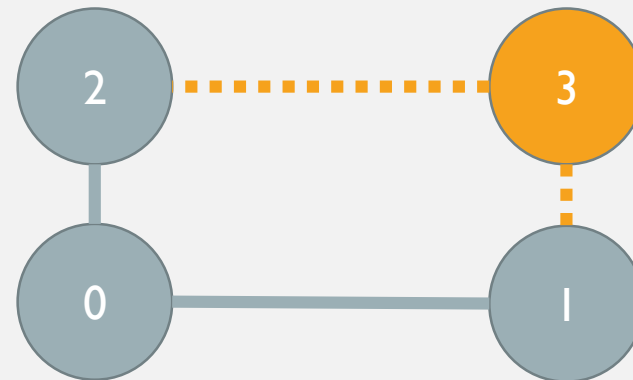
	0	1	2	3	4
0		Odd	Odd	Even	Even
1			Even	Odd	Odd
2				Odd	Odd
3					Even



# Solution

- Could be in either location
  - This becomes true for every node after this point

	0	1	2	3	4
0		Odd	Odd	Even	Even
1			Even	Odd	Odd
2				Odd	Odd
3					Even

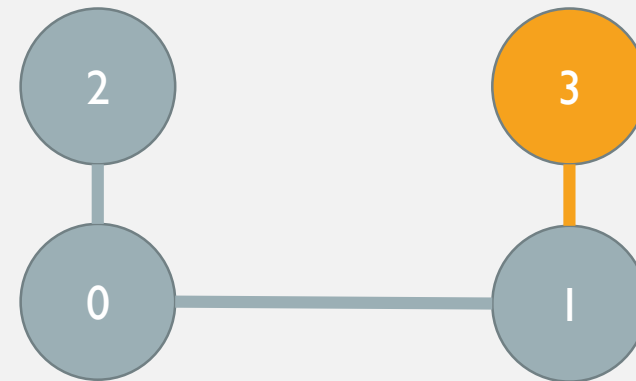




# Solution

- Default to the 'OddNode', node 1

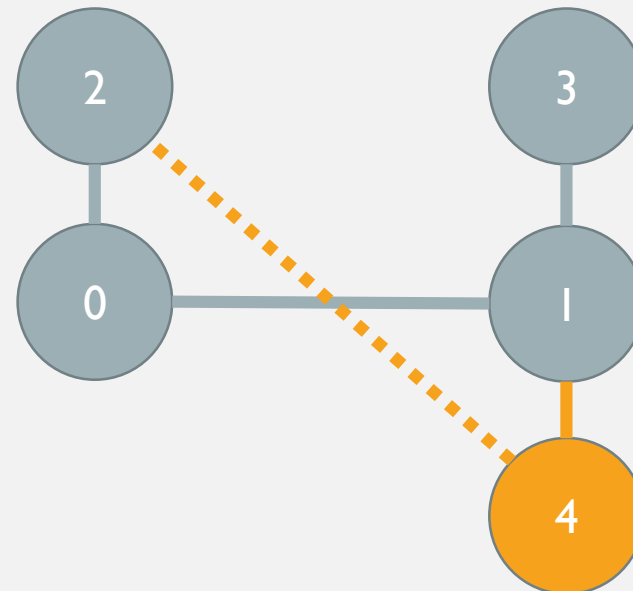
	0	1	2	3	4
0		Odd	Odd	Even	Even
1			Even	Odd	Odd
2				Odd	Odd
3					Even



# Solution

- Default to the 'OddNode' again

	0	1	2	3	4
0		Odd	Odd	Even	Even
1			Even	Odd	Odd
2				Odd	Odd
3					Even



# Solution

- Insight 4: Addition and Subtraction
  - +/- Two even numbers is an even number
  - +/- Two odd numbers is an even number
  - +/- One odd and one even number is an odd number
- Let's make everything relative to node 0

## Solution

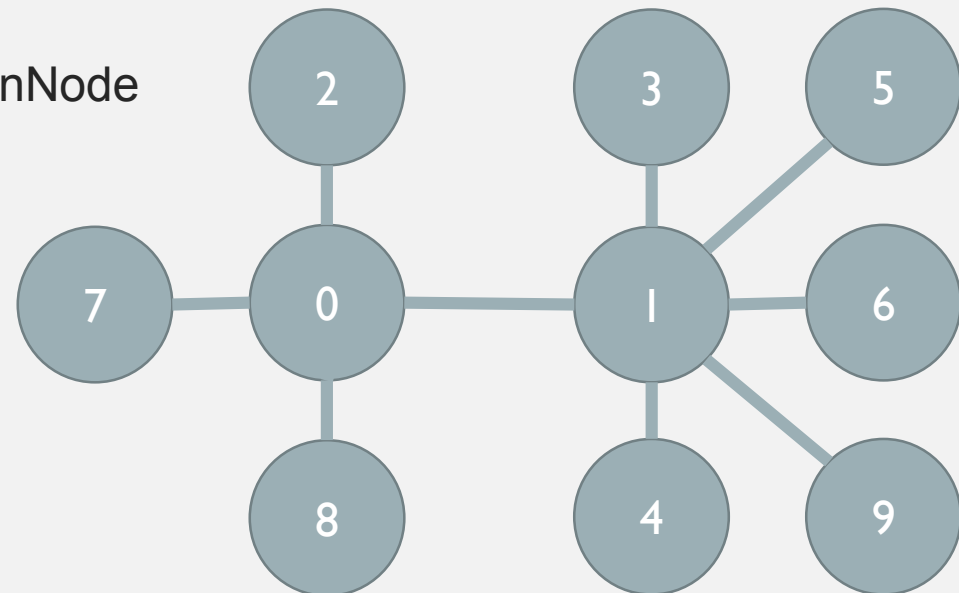
- Iterate over Node 0 as i
  - Iterate over Node 0 as j
    - i is odd, j is odd: (i,j) is even
    - i is odd, j is even (i,j) is odd
    - i is even, j is odd (i,j) is odd
    - i is even, j is even (i,j) is even

	0	1	2	3	4
0	Even	Odd	Even	Odd	Even
1	Odd	Even	Odd	Even	Odd
2	Even	Odd	Even	Odd	Even
3	Odd	Even	Odd	Even	Odd
4	Even	Odd	Even	Odd	Even

## Solution

	0	1	2	3	4	5	6	7	8	9
0	Even	Odd	Odd	Even	Even	Even	Even	Odd	Odd	Even

- Always use the OddNode and EvenNode

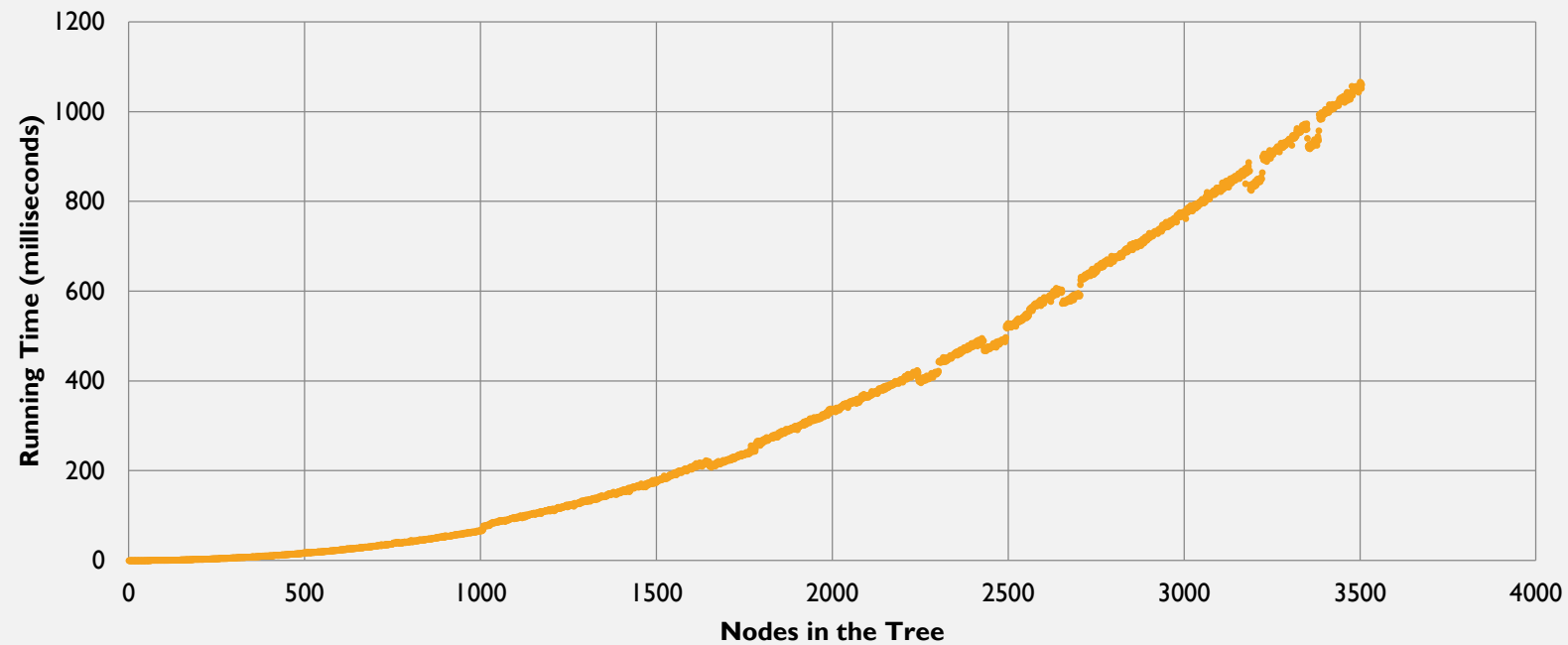


## Running Time

- Finding the evenNode:  $O(1)$
- Finding the oddNode:  $O(n)$
- Verifying the tree:  $O(n^2)$
- Creating the edge list:  $O(n)$
  
- $O(n^2)$  overall
  - N has a max of 50

# Performance

**Running Time in Milliseconds as Nodes in the Tree  
Increase (2.2 GHz Intel Core i7 Mac)**



## Other Solutions

- Must be at least  $O(n^2)$



## Topcoder results

- 497 Opens
- 470 Submits
- 314 Correct (66.81%)
- Best Time 4:54 C++
- Average Time 23:18
- Submission Accuracy Worst: Python (43%)
- Submission Accuracy Best: C# (80%)

Topcoder SRM 658, Division 1, 300 Points  
“OddEvenTree”

Elliot Greenlee  
EECS Department  
University of Tennessee

CS494 Presentation  
March 22, 2016