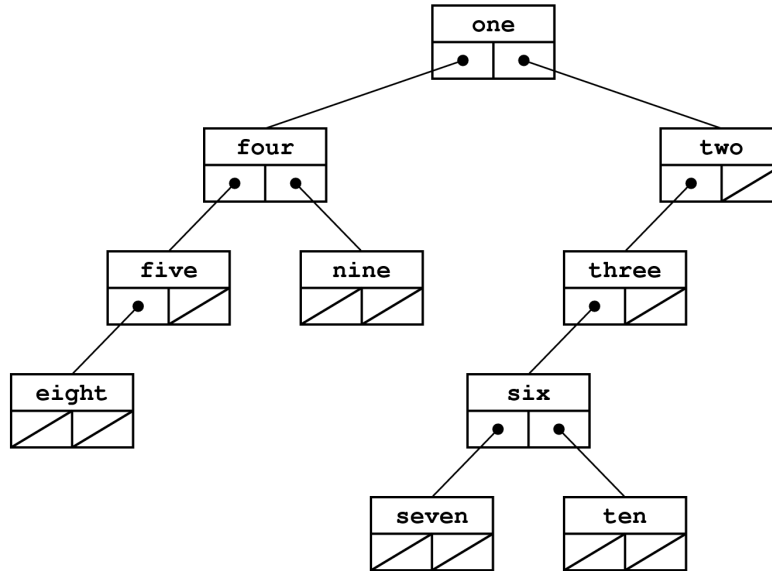


Solutions to Problem Set #2

Problem 1: Binary search trees

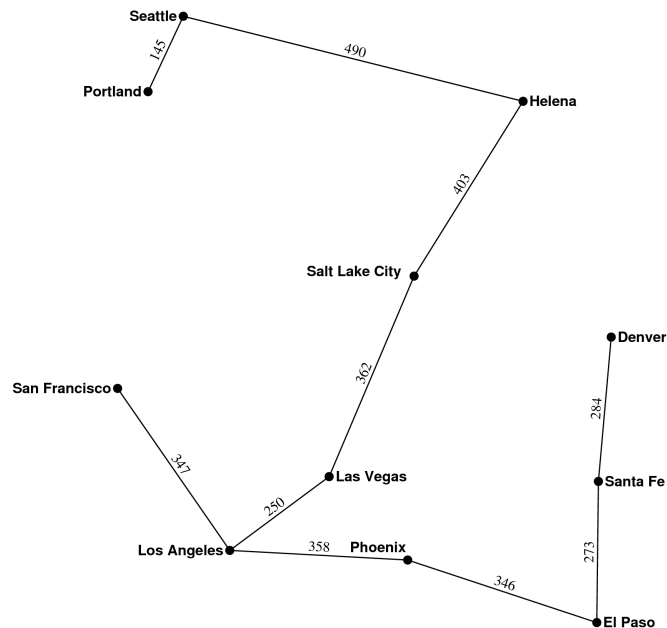


Problem 2: Graph traversals

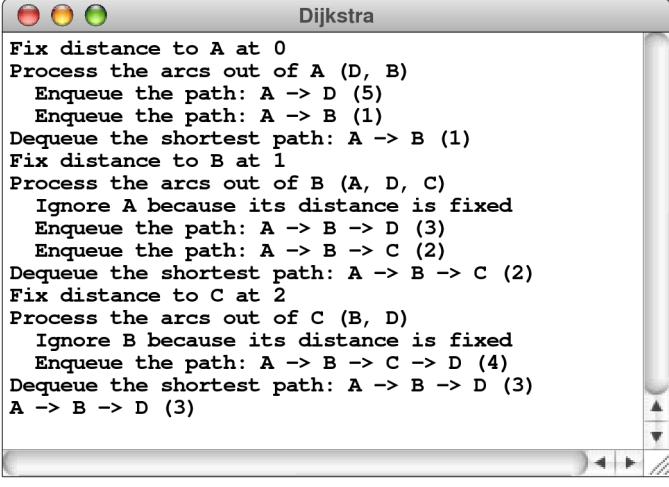
DFS from **STAN**: **STAN, SRI, UTAH, BBN, CMU, NRL, HARV, MIT, RAND, UCLA**

BFS from **CMU**: **CMU, BBN, NRL, UTAH, MIT, HARV, RAND, SRI, UCLA, STAN**

3. Minimum spanning trees



4. Dijkstra's algorithm



```

Dijkstra
Fix distance to A at 0
Process the arcs out of A (D, B)
  Enqueue the path: A -> D (5)
  Enqueue the path: A -> B (1)
Dequeue the shortest path: A -> B (1)
Fix distance to B at 1
Process the arcs out of B (A, D, C)
  Ignore A because its distance is fixed
  Enqueue the path: A -> B -> D (3)
  Enqueue the path: A -> B -> C (2)
Dequeue the shortest path: A -> B -> C (2)
Fix distance to C at 2
Process the arcs out of C (B, D)
  Ignore B because its distance is fixed
  Enqueue the path: A -> B -> C -> D (4)
Dequeue the shortest path: A -> B -> D (3)
A -> B -> D (3)
  
```

5. Expression trees

Parsing phase:

```

readE( ^y = 2 + ( x - 3 ) * 5 , 0)
readT( ^y = 2 + ( x - 3 ) * 5 )
readE( y = ^2 + ( x - 3 ) * 5 , 1)
readT( y = ^2 + ( x - 3 ) * 5 )
readE( y = 2 + ^ ( x - 3 ) * 5 , 2)
readT( y = 2 + ^ ( x - 3 ) * 5 )
readE( y = 2 + ( ^x - 3 ) * 5 , 0)
readT( y = 2 + ( ^x - 3 ) * 5 )
readE( y = 2 + ( x - ^3 ) * 5 , 2)
readT( y = 2 + ( x - ^3 ) * 5 )
readE( y = 2 + ( x - 3 ) * ^5 , 3)
readT( y = 2 + ( x - 3 ) * ^5 )
  
```

Evaluation phase:

```

IdentifierExp::eval(x) -> 11
ConstantExp::eval(3) -> 3
CompoundExp::eval((x - 3)) -> 8
ConstantExp::eval(5) -> 5
CompoundExp::eval(((x - 3) * 5)) -> 40
ConstantExp::eval(2) -> 2
CompoundExp::eval((2 + ((x - 3) * 5))) -> 42
  
```