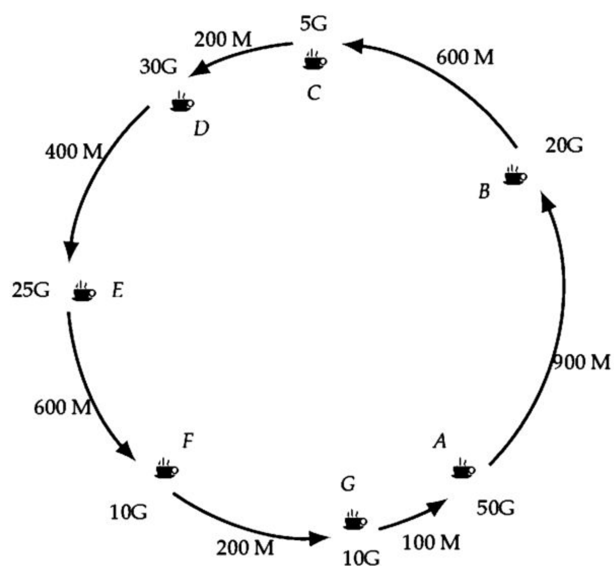
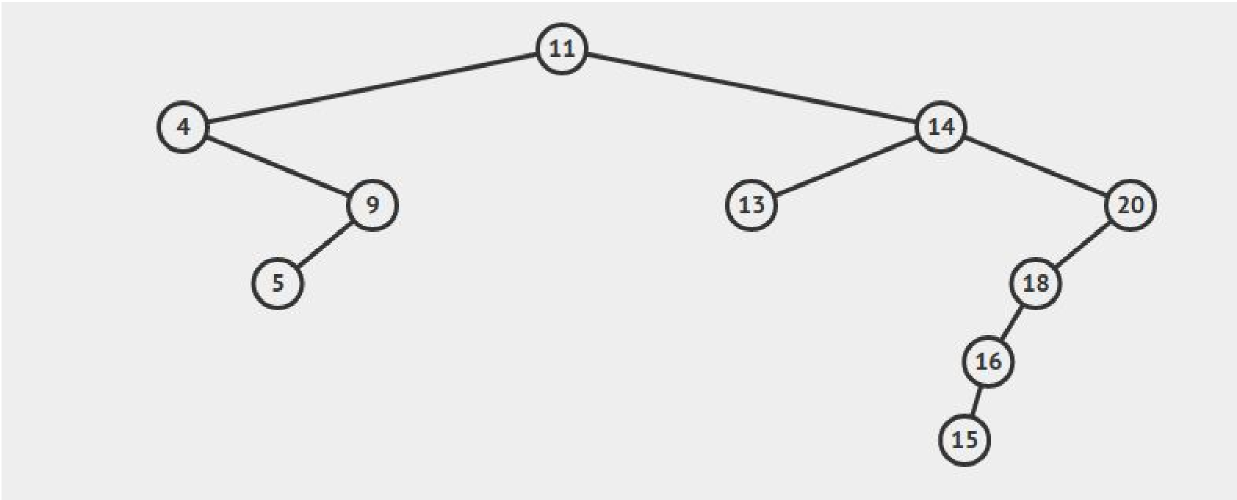
1. Let's say that you and a group of friends want to visit a bunch of cities and return to the first city in a specific order. However, your friends don't care which city you decide to start at. So for example, let's say that your friends want to visit the cities A, B, C, and D in that order. However, if you choose to start at city C, then you would visit C, D, A, B, and C. Each city has a gas station that specifies the most amount of gas you can take, and each city is some distance apart from each other. Let's say that each gallon allows you to drive 20 miles. Given the gas limit of each city as the first array, and the distance between each city as the second, find if there exists a starting city where you are able to go the entire trip without ever running out of gas (your tank may be empty when you reach the city).

   Let's say the input is as follows: [50, 20, 5, 30, 25, 10, 10], [900, 600, 200, 400, 600, 200, 100]. You can use the image below as a visual reference for what this would look like (First element of the first array is the maximum gallons one can get from city A, and the first element of the second array is the distance from city A to city B). Also note that if you start at city D, you are able to make the entire trip:



2. Perform a series of rotations so that you can recolor the BST, pictured below, into a red-black



3. We have a list of points on the plane. Find the K closest points to the origin (0, 0). (Here, the distance between two points on a plane is the Euclidean distance). You may return the answer in any order. The answer is guaranteed to be unique (except for the order that it is in). The algorithm should exhibit the optimal worst-case running time.

   Example:

   Input: points = [[3,3],[5,-1],[-2,4]], K = 2

   Output: [[3,3],[-2,4]]

   (The answer [[-2,4], [3,3]] would also be accepted.)

4. Explain why the worst-case running time for bucket sort is $\Theta(n^2)$. What simple change to the algorithm preserves its linear average-case running time and makes its worst-case running time O(nlg(n))?

5. Given a sequence of integers, write an algorithm that will find the length of the longest non decreasing subsequence. Let's say that you're given the following: [0, 8, 4, 12, 4, 2, 10, 6, 14, 1, 9]. The longest subsequence would be [0, 4, 4, 10, 14], which is a length of 5. *Note: The longest subsequence may not necessarily be unique*.

6. Suppose we perform a sequence of n operations on a data structure in which the ith operation costs i if i is an exact power of 2, and 1 otherwise. Use aggregate analysis to determine the amortized cost per operation.

7. We say that two "filled" squares belong to the same component, if they share an edge (sharing a corner is not enough). In the example below, there are 7 components.

   The grid is represented as an n x n matrix M, where a filled square is represented by a 1 in the corresponding entry of the matrix, and an empty one with a 0. Design an algorithm that takes M as input and computes the size (the number of squares) of the largest component in the grid (in the example above, the answer is 9). Write down the pseudocode and analyze its run time.

Example:

1	0	0	0	1	1	0	1
1	0	1	0	0	0	0	1
1	0	1	1	1	0	0	1
1	0	0	0	1	0	1	0
1	1	1	0	0	0	1	0
1	1	0	1	0	0	1	0
0	0	0	1	0	0	1	0
1	1	0	1	0	0	0	0

8. Using a disjoint-set forest with union by rank and path compression, show the data structure that results and the answers returned by the FIND-SET operations in the following program.

```
for i = 1 to 16
    MAKE-SET(x_i)
for i = 1 to 15 by 2
    UNION(x_i, x_i+1)
for i = 1 to 13 by 4
    UNION(x_i, x_i+2)
UNION(x_1, x_5)
UNION(x_11, x_13)
UNION(x_1, x_10)
FIND-SET(x_2)
FIND-SET(x_9)
```

Assume that if the sets containing x_i and x_j have the same size, then the operation UNION(x_i, x_j) appends x_j's list onto x_i's list.

9. Describe and analyze an efficient algorithm to find the 2nd smallest spanning tree of a given weighted undirected graph G, that is, the spanning tree of G with the smallest total weight except for the minimum spanning tree.