Plan

A. Read Java
B. Play with Java Collection’s Interface, List, Set, and Map. Play with PriorityQueue to get top k elements of an unsorted array. Play with Java’s Iterable, Iterator, Comparable, Comparator.
C. Implement basic algorithms and data structures from scratch.
    a. Sort algorithms,
    b. Trees
    c. Graphs, BFS, DFS and topological sort
    d. Dynamic programming, simple rabbit jumping, simple robot moving
    e. Heap
    f. Binary tree and Binary search tree
D. One time CC150 in this sequence, Stack&Queue, Linked List, Tree&Map, Sort&Search, String&Array, DP&Recursion.
E. Start Leetcode in sequence of AC rate.

1. Implement Heap (minHeap or maxHeap)

```
interface Heap{
        void add(int x);
        int remove();
        int size();
}
```

2. Perfect Shuffle

   Given an array of poker cards Card[52], shuffle it perfectly (each result of 52! results should have same prob. to occur). Can you do this in place?

3. K-way merge, k sorted array, k sorted linked list, k list of iterators

Given a k*n matrix, each row is sorted, return a one dimensional array, which is sorted.

Given List<List<Integer>> lists, where each List<Integer> is sorted, return a sorted result list.

Given List<Iterator<Integer>>, where each Iterator<Integer> is sorted, return a sorted result Iterator<Integer>

4. Running median

   Given an infinite stream of integers, write a method to return the median so far.
   Follow up, what if the stream is so large that cannot be hold in the memory?

5. Running average

Given an infinite stream of number, write a method to return the average of all the numbers occurred so far.

Follow up, large data case.

6. Stack/Queue support O(1) getMin()

Implement a MinStack class supporting push(), pop(), and getMin() all in O(1) time, where getMin() returns the minimum element in the stack.

```
public class Stack{
        void push(int x);
        int pop();
        int getMin();
}
```

Follow up, implement MinQueue.

7. Design data structure supporting insert, delete and getRandom(), all in O(1) time

```
public class RandomSet{
        void insert(int x);
        void delete(int x);//remove element from the set
        int getRandom();//get a random element from the set
}
```

8. Majority element of an array

Given an array, return the majority element of the array.
Majority of an array is the element, which occurs more than half times.

9. 2D board, find shortest distance to all the guards

Given a 2-D matrix represents the room, obstacle and guard like the following

```
0 0 0
B G G
B 0 0
```

where (0 is a room, B is an obstacle, G is a Guard):
calculate the steps from a room to nearest Guard and set the matrix, like this

```
2 1 1
B G G
B 1 1
```

10. Longest substring with only 2 distinct characters

Given a string, find the longest substring, which only has two different characters.
For example,
input = "aabaddaa",
output = “addaa”

11. Given dictionary, find all groups of anagrams

Given a dictionary, Set<String> dict, return List<List<String>> result, where each List<String> is a group of anagrams. If a word does not have anagrams in the dictionary, don't count it.

For example,
dictionary = {"cat","tac","dog","god","fish"}
output = {{"cat","tac"},{"dog","god"}}

12. Determine if string T's anagram occurs in string S
boolean isSubstringAnagram(String S, String T)
For example,
input, S = "abcdXYZdcd", T = "YXZ", output, true
input, S = abcdXYZ, T = "XYZc", false

13. Sort array into a1 <= a2 >= a3 <= a4 ….
Given an unsorted array, return array in this wave-like form.
For example, A = [1,2,3,4]
output = [1,4,2,3]

14. Topological sort, application on sorting tickets, find root in a tree representing in edges

Xiaoming recent traveled many cities, and he has a lot of tickets. He does not remember the begining city and the ending city. He only has a list of tickets, like below

```
class Ticket{
        String fromCity;
        String toCity
}
```

Please help him reconstruct the trip, returning a list of cities from start to end, i.e. a list of string.
For example,
Given tickets, List<Ticket> tickets =
{(Durham->NYC),(SF->Seattle),(NYC->SF),(Seattle->Miami)}
output = {Durham, NYC, SF, Seattle, Miami}

15. Given dictionary of alien language, rebuild the sequence of letters
An alien gives your a dictionary from his planet. Fortunately, their language also has 26 English letters from a to z. Unfortunately, the order of the letters is not abcdefg...z. Given the List<String> dictionary, please return a List<Character> to reconstruct the letter sequence.
For example, dictionary = {"dog","dot","ao","zoo",so"}
output = {d,a,z,s}

16. Given BST, find two elements that their sum equals to target
Given the root of a binary search tree, TreeNode root, and an integer target,
find two nodes, x and y, so that x.val + y.val = target

17. Given read4K() method, implement readAny(int size)

Consider this file class

```
public class File{
        String content;
        int cursor;
        String read4k();
}
```

The variable content represent the contents of the file, and the cursor is the current posistion of the cursor. The method read4k() return content.substring(cursor,cursor+4096). Please implement a new method, which can return any size of the file sequentialy. The signature of the method is

```
String readAny(int size);
```

18. implement int cubicRoot(int x)

Similar to the leetcode squareRoot(int x), please implement a methods which can return the cubic root of x.

19. Swap ith and jth bit of an integer

Given an integer, please swap the ith and jth bit in place.

20. Reverse bits of an integer

reverse bits of an integer in place

21. Shortest Manhattan distance to K points

The Manhattan distance can be found here: http://en.wikipedia.org/wiki/Taxicab_geometry

Consider a map with only horizontal streets and vertical streets. People can only walk along the streets. k people are standing on intersections. Find a meeting intersection that all k people travel least to meet.

22. Quick select kth smallest element in an unsorted array

Implement the quick select algorithm in the “Big Dictionary”

23. Top k elements of an unsorted array, of n sorted array, from n machine

Given an unsorted array, find the top k elements.
Given n sorted arrays, find the top k elements.
Given n sorted arrays in n different machines, find the top k elements.

24. Find intersection of two linked list, find lowest common ancestor in BT

Linked list intersection, see here http://www.geeksforgeeks.org/write-a-function-to-get-the-intersection-point-of-two-linked-lists/

This problem is similar to the Lowest Common Ancestor in Binary tree

25. Harry Potter, smallest strength to reach right bottom corner

假设你是harry potter，在grid的左上角，你现在要走到右下角，grid中有正数也有负数，遇到正数表示你的strength增加那么多，遇到负数表示strength减少那么多，在任何时刻如果你的strength小于等于0，那么你就挂了。在一开始你有一定的初始的strength，现在问这个初始的strength最少是多少，才能保证你能够找到一条路走到右下角。每一步只能向右或者向下。

26. Find diameter of a tree
The diameter of a tree is defined as the longest distance between any two leaf nodes.

27. Find number of elements equal to target in a sorted array
Given a sorted array, find how many elements equal to the given target.

28. Given a dictionary, generate a word with length k with letters based on the prob. occurring in the dict.

29. Number of unique urls, in a huge log file. Or the first unique url in a huge log file

30. LRU cache, distributed LRU cache

31. Implement a hash table from scratch

32. Doubly linked list to BST, BST to doubly linked list

33. Build balanced BST from sorted array, sorted linked list

34. Given 2D array representing the height of a Continent, find the continent divide points such that water can flow to both Atlantic ocean and Pacific ocean.

35. Given a string, if any substring containing at almost three letters, it is called valid string. If the string can only consist A, B and C three letters. Write a method to compute how many different valid string with given length N.

36. Write a decorator for Iterator, supporting peek functionality.
37. K-window sum of an array, input = [1,2,3,4,5], k = 2, ouput = [3,5,7,9]
38. Given dictionary, and a string W, find the longest string in the dict that only using letters in W.
39. Given an array, return the product of all elements except for itself. For example, A = [1,2,3,4], B = [24,12,8,6]. Follow up, if A contains zeros. Follow up, you cannot use divide. Follow, how about if A is very long.
40. Given a set of coins with values, find number of ways to reach a sum of n. A coin can be used, only one time, k times, infinity times
41. Integer represent by an array, implement plus one.

42. Game of Life. Not so important.
43. Shortest path from source to destination in a 2D array, with blocks
44. Find the successor and the predecessor of a key in a BST
45. Given a list of parent-children relationship, build a graph, if it is a tree, find the root, and print the tree.
46. Given a list of intervals, where each interval has a start time and a end time, find a time with the most intervals. For example, the interval may represent the time when people entering and leaving the room, you need to find the time when there are most people in the room.
47. Given an array, find the number of subarrays that sum up to zeros. A problem of my onsite of Google
48. Write a class implement hasNext() and next() methods, its constructor will take a list of iterators,
49. Implement nextRandom(int N, int[] blackList), which return a random number between 1 to N, and not shown in blackList.
50. Given array and target, find number of triplets x, y, z that x+y+z <= target.
51. Find array, determine if there is a triplet, that i<j<k, and A[i] > A[j] > A[k].
52. Given a string containing question mark, return a possible output that replace question mark by 0 or 1. For example, input = “a?b?c”, output = [“a0b0c”,”a0b1c”,”a1b0c”,”a1b1c”]
53. Stack support getMostFrequentPushedItem()
54. Write a graph class, node class, edge class, implement bfs, dfs and topological sort.
55. Given N, print all prime numbers smaller than N.
56. Cut log using dynamic programming
57. Fibonacci numbers, recursive, iterative, dynamic programming, and matrix solution
58. Ugly numbers
59. Flatten a linked list,

```
class ListNode{
        int val;
        ListNode next;
        ListNode nextLevelHead;
}
```

60. Find next higher number with same digits.
    Example 1 : if num = 25468, o/p = 25486
    Example 2 : if num = 21765, o/p = 25167
    Example 3 : If num = 54321, o/p = 54321

61. Large Scale Data Manipulation, http://blog.csdn.net/v_july_v/article/details/7382693
62. A file with unknown number of lines, return a random line of it.
63. Optional data structure, Interval tree, skip list, trie,
64. Optiinal advanced algorithms, especially for the string matching, KMP, RK, Finite Automate

65. Design a system to return top k goole queries
66. Design Google calendar
67. Design Google suggestion
68. Design a Rate Limiter
69. Design distributed key-value store