**Search**

279. Perfect Squares

a. BFS

1. get the possible square numbers

2. update the number to check and increases the count(stack)

3. return when the number is equal to sqrt number

b. DP

1. get the possible square numbers

2. the current least number is the min–term of num[cur – sqrt] + 1

3. return num[n]

127. Word Ladder

a. BFS

1. make a letter empty and get the possible words in a dict

2. update the current transformed word and increases the count(stack)

3.return the number when it is equal to the endWord

695. Max Area of An Island

a. DFS

1. iterate all values and check if it’s a one

2. if true, mark it with 0 and increases the area by one

3. then do depth-first-search to its adjacent values

200. Number of Islands

a. DFS

1. iterate all values and check if it’s a one

2. if true, increases the count

3. mark it with 0 and do depth-first-search to all its adjacent values

547. Friend Circles

a. DFS

1. iterate all people and check his relationship array

2. increase the circle by one

3. mark all his friends and do depth-first-search to them

130. Surrounded Regions

a. DFS

1. mark all ‘O’ regions with ‘S’ and do depth-first-search to its adjacent regions

2. convert all ‘S’ regions to ‘X’ and everything else to ‘O’

417. Pacific Atlantic Water Flow

a. DFS

1. do depth-first-search to the margin areas for both oceans

2. list the regions that can flow to both oceans

17. Letter Combinations of a Phone Number

a. back-tracking

1. map the letters with the numbers

2. do backtracking to the letters and append it when the length of letters is equal to zero

93. Restore IP address

a. back-tracking

1. the current number can be the first one, two, three digits

2. do back-tracking for the rest digits