**Assignment 16**

44.

class Solution {

    public boolean isMatch(String s, String p) {

        int m = s.length();

        int n = p.length();

        boolean[][] dp = new boolean[m+1][n+1];

        dp[0][0] = true;

        for (int j = 1; j <= n; j++) {

            if (p.charAt(j-1) == '\*')

                dp[0][j] = dp[0][j-1];

        }

        for (int i = 1; i <= m; i++) {

            for (int j = 1; j <= n; j++) {

                char sc = s.charAt(i-1);

                char pc = p.charAt(j-1);

                if (pc == '?' || pc == sc) {

                    dp[i][j] = dp[i-1][j-1];

                } else if (pc == '\*') {

                    dp[i][j] = dp[i][j-1] || dp[i-1][j];

                } else {

                    dp[i][j] = false;

                }

            }

        }

        return dp[m][n];

    }

}

//673

class Solution {

    public int findNumberOfLIS(int[] nums) {

        int n = nums.length;

        if (n == 0) return 0;

        int[] lengths = new int[n];

        int[] counts = new int[n];

        for (int i = 0; i < n; i++) {

            lengths[i] = 1;

            counts[i] = 1;

        }

        int maxLen = 1;

        for (int i = 1; i < n; i++) {

            for (int j = 0; j < i; j++) {

                if (nums[j] < nums[i]) {

                    if (lengths[j] + 1 > lengths[i]) {

                        lengths[i] = lengths[j] + 1;

                        counts[i] = counts[j];

                    } else if (lengths[j] + 1 == lengths[i]) {

                        counts[i] += counts[j];

                    }

                }

            }

            maxLen = Math.max(maxLen, lengths[i]);

        }

        int result = 0;

        for (int i = 0; i < n; i++) {

            if (lengths[i] == maxLen)

                result += counts[i];

        }

        return result;

    }

}