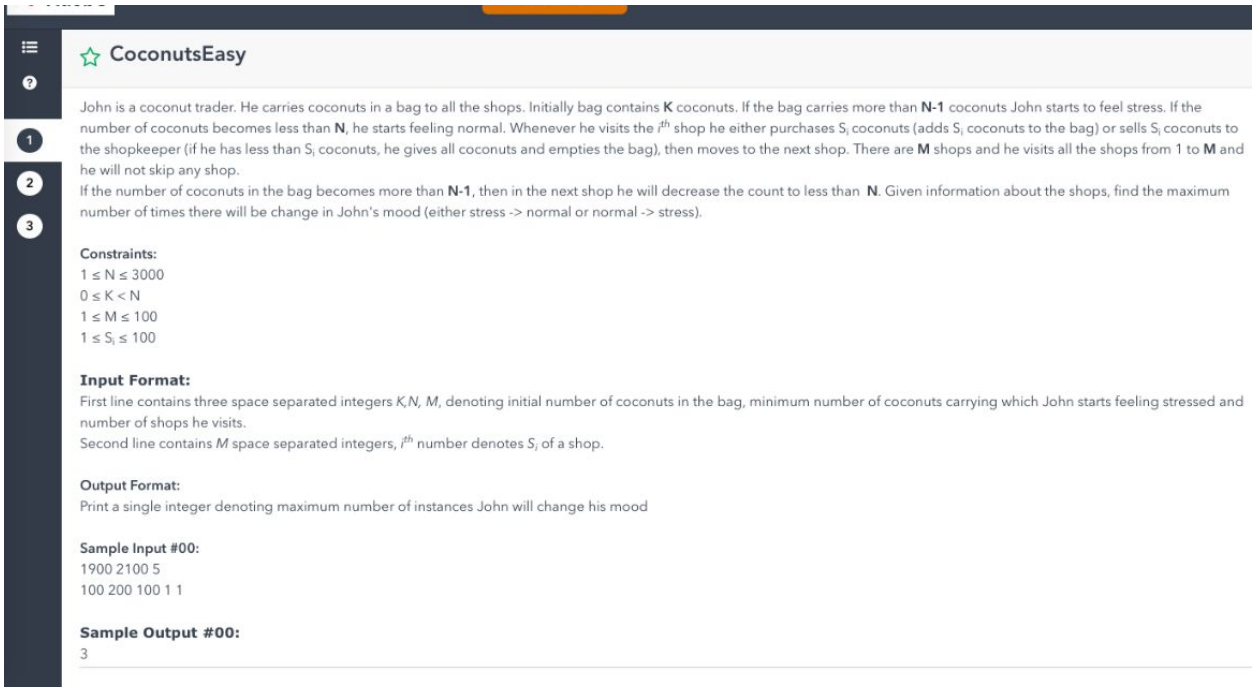
Adobe Questions:

1. Sort the characters of a string and print the string?
   https://www.geeksforgeeks.org/sort-string-characters/

2. find the lexicographically smallest string after rotating a string?
   https://www.geeksforgeeks.org/lexicographically-minimum-string-rotation/

3.



CoconutsEasy

John is a coconut trader. He carries coconuts in a bag to all the shops. Initially bag contains **K** coconuts. If the bag carries more than **N-1** coconuts John starts to feel stress. If the number of coconuts becomes less than **N**, he starts feeling normal. Whenever he visits the i^{th} shop he either purchases S_i coconuts (adds S_i coconuts to the bag) or sells S_i coconuts to the shopkeeper (if he has less than S_i coconuts, he gives all coconuts and empties the bag), then moves to the next shop. There are **M** shops and he visits all the shops from 1 to **M** and he will not skip any shop.
If the number of coconuts in the bag becomes more than **N-1**, then in the next shop he will decrease the count to less than **N**. Given information about the shops, find the maximum number of times there will be change in John's mood (either stress -> normal or normal -> stress).

Constraints:
$1 \le N \le 3000$
$0 \le K < N$
$1 \le M \le 100$
$1 \le S_i \le 100$

Input Format:
First line contains three space separated integers *K,N, M*, denoting initial number of coconuts in the bag, minimum number of coconuts carrying which John starts feeling stressed and number of shops he visits.
Second line contains *M* space separated integers, i^{th} number denotes S_i of a shop.

Output Format:
Print a single integer denoting maximum number of instances John will change his mood

Sample Input #00:
1900 2100 5
100 200 100 1 1

Sample Output #00:
3

http://qa.geeksforgeeks.org/4323/qa.geeksforgeeks.org/4323/finding-maximum-number-of-mood-swings.html?fbclid=IwAR17pY87_Dq2jM934XaSEH3T7jczsKfU5fn3YpqQXH5HrULWjR1k_7c9QWc

4. https://www.geeksforgeeks.org/queries-counts-array-elements-values-given-range/
5. https://www.geeksforgeeks.org/minimum-sum-absolute-difference-pairs-two-arrays/

6.
A square grid(matrix - n x n) is given, with values 0, 1 or -1 in each cell. 0 means there is a path via that square. 1 means there is a diamond in it(and obviously a path via it). -1 means no path, i.e. obstacle. Starting from (0,0) u need to go to (n-1,n-1) and return back again. When going from (0,0) to (n-1,n-1) you can take only right or down and during returning u can only take left or up. The question is to find maximum number of diamonds that can be collected in a round trip.If no path exists return 0(since u collected no diamond)
https://www.geeksforgeeks.org/maximum-points-top-left-matrix-bottom-right-return-back/

7. .https://www.hackerrank.com/challenges/challenging-palindromes/problem

Ex: given: 3
ban
3
ana
Output: 5 (because of anana)
Exp: palindromic string S will be an + ana = anana.

Solution: concat s1 , s2 lets say s3 = s1+s2, apply LCS on reverse of(s3)(row) and s3(column) , bottom rightmost value, will be the answer.

8.
Input 1: babababa
Output 1: ababa
Input 2: bababababab
Output 2: babab
Solution: take an unordered map(string, int) and put all substring of length 5 in it, sort it , now traverse the map, the first maximum count(frequency) value will correspond to largest frequent substring of length 5.

9. Ex: given array = 11223
Output: 3
Exp: Remove 11 and then 22. Only 3 is left in the array.
Given array = 21123
output : 3
Exp: remove 11, then array will be 223, now remove 22.
Given array = 21132
Output: 232
Exp: remove 11, then array will be 232. Can't remove further.

```
vector<int> remDigit(vector<int>&num,int n)
{
   stack<int>s;
   s.push(num[0]);
   for(int i=1;i<n;i++)
   {
        if(s.top()!=num[i] && s.size()>1)
        {
             int x=s.top();
             s.pop();
             if(!s.empty() && s.top()!=x)
```

```
                    s.push(x);
                else
                {
                    while(!s.empty() && s.top()==x)
                        s.pop();
                }
            }
            s.push(num[i]);
        }

        if(s.size()>1)
        {
            int x=s.top();
            s.pop();
            if(!s.empty() && s.top()!=x)
                s.push(x);
            else
            {
                while(!s.empty() && s.top()==x)
                    s.pop();
            }
        }
        vector<int>res;
        while(!s.empty())
        {
            res.insert(res.begin(),s.top());
            s.pop();
        }
        return res;
    }
```

Samsung Questions

1. Cycle in directed graph number of vertex(n), number of edge(m). Then in next line m pairs of numbers representing edges of directed graph. Find if there is some cycle. If yes, print cycle in ascending order of vertex numbers involved in the cycle else print 0 (if there are multiple cycles print any one)

   https://www.geeksforgeeks.org/detect-cycle-in-a-graph/

2. 2-coloring in Undirected graph Given an undirected graph, if the graph can be coloured in two colors such that no two adjacent vertices are of same color then print the vertices which belong to the same color (you can print vertices with color 0/1), otherwise print -1.

Input : First line gives number of vertices(V) and edges(E) (e.g. 7 10)
Next line contains E pairs representing edges.
https://www.geeksforgeeks.org/bipartite-graph/
If bipartite then print the vertices from vector colorArr which belongs to same color.

3. https://github.com/ABHISHEKVALSAN/PLACEMENT-PREP/blob/master/TESTS/Samsung/rockClimbing.cpp

4. Mr. Kim has to deliver refrigerators to N customers. From the office, he is going to visit all the customers and then return to his home. Each location of the office, his home, and the customers is given in the form of integer coordinates (x,y) (0≤x≤100, 0≤y≤100) . The distance between two arbitrary locations (x1, y1) and (x2, y2) is computed by |x1-x2| + |y1-y2|, where |x| denotes the absolute value of x; for instance, |3|=|-3|=3. The locations of the office, his home, and the customers are all distinct. You should plan an optimal way to visit all the N customers and return to his home among all the possibilities.

https://ideone.com/fork/wUhGKM

5. Initially you have H amount of energy and D distance to travel, you are given two arrays of size 5, each indicating the amount of energy you can utilise and the time it will take to cover the next distance with that energy. For eg.
h(array of energy) = { 4 , 5 , 7 , 12, 2 }
t(corresponding array of time) = {5min20sec, 3min20sec, 2min30sec, 1min0sec, 15min20sec}
You start at 1 and you need to take one of the five energies to move to 2 and so on until you reach D. The task is to find the d minimum time required for given H,D,h and t.
Question reworded: Initially you have H(<4000) amount of energy and D(<1000 km) distance to travel. You may travel with any of the given 5 speeds. But you may only travel in units of 1 km. For each km distance travelled, you will spend corresponding amount of energy. E.g. the k;given speed are :

Cost of travelling 1 km:	4	5	2	3	6
Time taken to travel 1 km:	200s	210s	230s	235s	215s

The task is to find minimum time required to cover total D km with remaining H >= 0.

https://github.com/kaushal02/interview-coding-problems/blob/master/energyDifference.cpp

6. There is one spaceship. X and Y co-ordinate of source of spaceship and destination spaceship is given. There are N (0<=N<=5) number of wormholes each wormhole has 5 values. First 2 values are starting co-ordinate of wormhole and after that value no. 3 and 4 represents ending co-ordinate of wormhole and last 5th value is represents cost to pass through this wormhole. Now these wormholes are bi-directional. Now to go from (x1,y1) to (x2,y2) is

abs(x1-x2)+abs(y1-y2). The main problem here is to find minimum distance to reach spaceship from source to destination co-ordinate using any number of wormhole. It is ok if you won't use any wormhole.

https://ide.geeksforgeeks.org/pC9w4ETP2x

7. Burst Balloons:

Given n balloons, indexed from 0 to n-1. Each balloon is painted with a number on it represented by array nums. You are asked to burst all the balloons. If the you burst balloon i you will get nums[left] * nums[i] * nums[right] coins. Here left and right are adjacent indices of i. After the burst, the left and right then becomes adjacent.
Find the maximum coins you can collect by bursting the balloons wisely.

```
int maxCoins(vector<int>& nums) {
        int n  = nums.size();
        /// Insert two 1s in the begining and in the end.
        nums.insert(nums.begin(), 1);
        nums.push_back(1);

        vector<vector<int>> c(n+2, vector<int>(n+2, 0));
        for(int l  = 1; l <=n ; ++l) {
        for (int i = 1; i <= n-l+1; ++i) {
                int j = i + l -1;
                for(int k = i; k <= j; ++k) {
                c[i][j] = max(c[i][j], c[i][k-1]+ nums[i-1]*nums[k]*nums[j+1] + c[k+1][j]);
                }
        }
        }
        return c[1][n];
        }
```

8. You have a matrix of 0 and 1 of order N*M and a parameter K is given. You have to perform the operation of flipping any column of matrix exactly K times. Flipping means changing 0 to 1 and 1 to zero. This operation can be performed any number of times on the same column . Using this operation, maximize number of rows filled with all 1. First line is number of test cases, next line is N, M and K, and then N*M matrix follows.
https://stackoverflow.com/questions/7116438/algorithms-question-flipping-columns?fbclid=IwAR212Mh6pR0JeXxj3IX0j-Js-TXSbLoq44cx0mlgg0e1D3EFck68xFV8e7o

9.
https://www.geeksforgeeks.org/samsung-semiconductor-institute-of-research-ssir-software-intern-fte-set-1/?fbclid=IwAR212Mh6pR0JeXxj3IX0j-Js-TXSbLoq44cx0mlgg0e1D3EFck68xFV8e7o

10. New common question Soln