Day 3: Problem Analysis

16.04.2015

Problem A. North North West

- You are given a string, that describes the direction using words "north" and "west"
- You need to calculate the rational number — the angle corresponding to that direction
- Sum up from right to left: $d_0 + 90 \cdot \sum_{i=1}^{n-1} \frac{a_i}{2^i}$, where a_i is -1 or 1 and d_0 is 0 or 90, depending on last element of sequence

Problem B. Unknown Switches

- There are switches and light bulbs, each light bulb can be turned on or off by exactly one switch
- Some sets of switches were triggered and states of bulbs are given, for every bulb find which switch controls it
- Find the answer for every bulb independently
- If the bulb state was flipped, then the set of switches contains controlling switch
- If wasn't, then complementary set contains it
- Intersect all this sets, if there is exactly one element, then it's the answer, otherwise output "?"

Problem C. Speedrun

- $n+1$ points given on line, in one minute player can pass from $(i-1)$-th point to i-th with probability p_i
- Player can save his progress, it takes one minute
- So with probability $1-p_i$ player moves to last saved point
- Find the expected time need to get from 0 to n-th point, if player plays optimal

Slow solution

- It's always needed to remember only the last saved point, it doesn't matter what points player has saved before
- One can compute dynamic programming $f[i]$ — expected time to get from i-th point to n-th, if he saved at point i
- $f[i] = 1 + \min\limits_{j>i}(f[j] + E(i,j))$, where $E(i,j)$ — expected time to get from i to j without saving his progress
- $E(i,j) =$
$$E(i,j)(1 - \prod_{k=i+1}^{j} p_k) + 1 + p_{i+1} + p_{i+1}p_{i+2} + \ldots + \prod_{k=i+1}^{j-1} p_k$$
- $E(i,j) = \frac{1+p_{i+1}+p_{i+1}p_{i+2}+\ldots+\prod\limits_{k=i+1}^{j-1} p_k}{\prod\limits_{k=i+1}^{j} p_k} = E(i+1,j) + \frac{1}{\prod\limits_{k=i+1}^{j} p_k}$

Solution

- That solution works in $O(n^2)$
- It is useless to save in point $i-1$ if $p_i = 1$, so one can just remove all this points, adding weights to the edges
- One could see that if $\frac{1}{\prod\limits_{k=i+2}^{j} p_k}$ is more than 2 it is faster to save in $(i+1)$-th point and then go to j
- Since $p_i \leqslant 0.99$ then $0.99^{70} < 0.5$, so only 75 transitions are enough to try from every point

Problem D. Flowers

- There are n flowers, we can water all the plants by W water and fertilize each plant with its own fertilizer
- There is the cost for a unit of water and cost for a unit of each fertilizer
- For every plant it's known the vitality added by one unit of water and vitality added by one unit of fertilizer
- Find minimal cost needed so that vitality of each plant exceeds its limit
- If we fix W, then we can find the cost needed to make plant happy
- That cost is piecewise linear of W
- For every plant make events when linear function changes, and sum of linear functions is linear function
- Use scanline and find the minimum of linear function at each step

Problem E. Square in Circles

- There are circles given, center of each lying on x-axis
- Find the largest square, with sides parallel to axes, such that every point square lies inside of some circle
- Because of the symmetry center of square lies on x-axis
- Do binary search on the side of the square, let's check: is it possible to get square with side length L
- Two sides of this square lie on lines $y = \frac{L}{2}$ and $y = -\frac{L}{2}$
- Intersect all the circles with line $y = \frac{L}{2}$, if there is the segment of length at least L in union of these intersections, then such square exists

Problem F. Reverse a Road II

- Directed graph is given, you can reverse at most one road, such that the number of edge-disjoint paths is maximal
- Number of edge-disjoint paths is minimal-cut, as well as maximal flow
- Find max-flow. Let S be the set of vertices, that are reachable from s in residual network, and T be the vertices, that are reachable from t by reversed edges in residual network
- If there exists edge in residual network from T to S, then one can reverse it and maximal flow increases by one

Problem G. Cookie Counter

- Given N, D and X. Find the number of sequences a of length D, such that $\sum\limits_{i=1}^{D} a_i = N$ and $a_i < X$, for every i
- $N, X \leqslant 2000$ and $D \leqslant 10^{12}$
- Use inclusion-exclusion principle. Let A_i be the set of sequences, such that $a_i \geqslant X$
- Answer is equal to $|\overline{A_1 \cup A_2 \cup \ldots \cup A_D}|$
- $|\overline{A_1 \cup A_2 \cup \ldots \cup A_D}| = \binom{N+D-1}{D-1} - \sum\limits_{i=1}^{N} \binom{D}{i} \binom{N-i \cdot X+D-1}{D-1}$

Problem H. Points and Lines

- The problem was to parse the input and make simple geometry functions

Problem I. Color the Map Extreme

- Given planar graph, find its chromatic number
- It's well known that planar graph can always be painted in 4 colors
- So one needs to check if chromatic number equals to 1, 2 or 3
- 1 and 2 can be done easily in polynomial time
- To check if it's 3, problem authors advise to read some papers about 3-coloring in the internet

Problem J. Website Tour

- Given a directed graph, you walk on the graph, when coming to vertex you can spend t_v time to get p_v points, and in vertex v you can do that at most k_v times
- Find the maximal number of points you can get
- Firstly, find strongly connected components and learn for every vertex is it reachable from itself, if it's not then make $k_v = 1$
- Compute dynamic programming: $f[v][t]$ — maximal number of points you can get, starting at component v and spending t units of time
- You can either go to another component, then $f[v][t] = \max\limits_{u} f[u][t]$
- Or you can walk in current component, then use knapsack idea to update DP

Problem K. Idempotent Filter

- Every cell in hexagonal grid was painted black or white
- Function could be applied, every cell's new value depends on the old values of its neigbours
- Function given, find if it's idempotent or not
- One need to make two operations and find out, that the value of the cell doesn't change after second application, no matter what cells are around
- The value of the cell after two moves depends on the value of 19 cells, that are at distance up to 2
- Try all the values of those cells and make two applications of function, check for every of them