

比赛链接: <https://ac.nowcoder.com/acm/contest/303>

比赛花絮&&现场赛情况: (网差可能进不去)

<http://tokitsukaze.live/2018/12/08/2018NIT15thProgrammingContest/>

(PS: 欢迎大家参加牛客练习赛 34, 出题人 tokitsukaze、winterzz1)

出题: tokitsukaze, winterzz1, teitoku

难度预期:

(1)对于新生:

easy: AJIB

medium: DEK

hard: HFL

very hard: CG

(2)对于牛逼网友:

easy: ABDEFHIJKL

medium: C

hard: G

A.StarCraft

Author: tokitsukaze。

ans=n-20+2018

B.Fibonacci and Counting

Author: winterzz1。

ans=n+1

证明:

定理: 任意相邻两项的斐波那契数互质。

假设 $F(n)$ 与 $F(n+1)$ ($n > 2$) 有公约数的话, 不妨设为 a , 应有 a 大于 1。那么再根据 $F(n+1) = F(n) + F(n-1)$, a 应能整除 $F(n-1)$, 即 $a | F(n-1)$, 再结合 $a | F(n)$, $a | F(n+1)$, 可知 $a | F(n-2)$, 以此类推, 我们会发现 $a | F(2)$ 和 $a | F(1)$, 而 $F(1) = F(2) = 1$ 。这是不可能的, 假设不成立。所以斐波那契数列邻近项互质。

根据辗转相除法, $\gcd(a, b) = \gcd(b, a \% b)$, 而斐波那契数的后一项不大于前一项的 2 倍, 所以这里的模其实就是减法。也就是: $\gcd(a, b) = \gcd(b, a - b)$ 。

观察一下就会发现, 这个式子就是斐波那契数列运算的逆运算。所以使用几次斐波那契运算过去, 就得使用几次辗转相除回来, 再算上第一次调用的一次, 答案为 $n+1$ 。

C.LCPS

Author: tokitsukaze。

把第一个串建回文树, 回文树的每个节点记录下第一次出现的下标。

第二个串在回文树上匹配, 匹配时更新最大长度, 并记录下出现的下标。

然后把子串暴力取出来, 直接比较字典序即可。因为本质不同的回文串是 $O(n)$ 的, 所以卡不掉暴力比较字典序。

D.Campaign

Author: teitoku。

二进制枚举答案, 然后维护一个可行的 l, r , 当可行的 l, r 包含了输入的个数的时候, 就是满足条件的, 然后从所有答案中选出最大的就行。

E. Build Pylons

Author: winterzz1。

题意：现在有个人在数轴上从左到右依次经过一些点，他每走一步时的花费都是等差数列 1,3,5,7,9... 然后一旦他经过某个点等差数列就会从 1 开始重新计数。

根据等差数列求和公式可得走 k 步的花费总共为 k^2 ，所以先对输入的 a 数组进行排序。然后依次计算 $sum += (a[i] - a[i-1]) * (a[i] - a[i-1]) (i > 1)$ 即可。最后再加上最后一个建筑的建造时间就是答案。

F. Pylon Link

Author: tokitsukaze。

solution (1): 二分答案，check 连通性。 $O(\log(1.5e9) * n^2)$ 。

solution (2): n^2 建图，求最小瓶颈生成树。prim: $O(n^2)$ ，kruskal: $O(\log(n^2) * n^2)$ 。

G. Rubik's Cube

Author: winterzz1。

三阶魔方翻楞公式: $MU^2M'UMU'M'$ (F 面下侧楞块翻转，D 面其他块不变)

三阶魔方翻角公式: $RUR'U'RUR'$ (F 面右下角块顺时针旋转单次，D 面其他块不变)

所以只要求在一个面上拼数字的话，只要数字不在同一个块中，就一定能够放在同一面上。

一个魔方由 8 个角块，6 个中心块，12 个棱块组成。

拼好的一个面上有 4 个角块，4 个楞块，1 个中心块。

所以使用分组背包处理一个块中不同面的互斥关系，然后从 8 个角块中选 4 个，12 个棱块中选 4 个，6 个中心块中选 1 个放入背包。

H. Protoss and Zerg

Author: tokitsukaze。

加法原理与乘法原理，需要会快速幂。(解法在样例解释就给出了)

I.Race Sorting

Author: tokitsukaze。

开几个数组直接扫，按题意模拟即可。

J.Carrier

Author: tokitsukaze。

if else 讨论一下，过样例应该就能过。

K.Technology Tree

Author: tokitsukaze。

建树直接 dfs 即可。

L.The Last Stand

Author: winterzz1。

题意：从最左边走到最右边，每个物品可以取或者不取，取东西的时候首先会获得一个 val 权值，然后接下来会持续的获得或者减少该物品 delta 属性的权值。

dp[i]表示取第 i 个物品能获得的最大权值。然后设置两个虚点 0 和 n+1，这两个虚点的 val 属性和 delta 属性均为 0

dp 转移方程为： $dp[i]=\max(dp[i],dp[j]+\delta[j]*(pos[i]-pos[j])+val[i]);(i>j)$

注意条件要求过程中权值不能为负，所以需要先判断是否可以转移再进行状态转移。