

atcoder Alsing Programming Contest 2020

链接 <https://atcoder.jp/contests/aising2020/>

A Number of Multiples

- 题意：闭区间 $[L,R]$ 中有多少个数字 d 的倍数
- 题解：大水题，直接遍历

B An Odd Problem

- 题意：给出编号从 1 到 N 的一串数字，问满足编号与数字本身都是奇数的数字有多少个
- 题解：大水题，直接遍历

C XYZ Triples

- 题意：定义 $f(n)$ 是满足下列条件的三元组 (x,y,z) 的数量：
 - $1 \leq x,y,z$
 - $x^2 + y^2 + z^2 + xy + yz + zx \leq n$
- 题解：水题，三重循环遍历，用一个数组存每个数字的结果(直接暴力就行)

D Anything Goes to Zero

- 题意：有一个长度为 N 的二进制字符串(由 $0,1$ 组成) 定义 $f(i)$ 为进行如下操作的次数，并对每个 i 输出 $f(i)$
 - 操作：用 x_i 表示二进制字符串第 i 位取反后的十进制数，用 $\text{popcount}(x_i)$ 代表目前二进制表示中 1 的数量，用 $x_i \% \text{popcount}(x_i)$ 直至其值为 0 时结束
- 题解：记 x_i 中 1 的个数为 tot 若第 i 位为 1 tot 减 1 ，反则加 1 ，然后分别计算 $\text{tot}-1$ 与 $\text{tot}+1$ 的情况，之后再计算每一位的时候进行加减就行，若为 0 ，那么加上 2^i 再对 $\text{tot}+1$ 取模，若为 1 ，则减去 2^i 再对 $\text{tot}+1$ 取模。

E Camel Train

- 题意：有序号为 $1,2,\dots,N$ 的 N 只骆驼，现需要对骆驼进行排序，如果第 i 只骆驼排序后是前 K_i 只骆驼，那么它的幸福度为 L_i 否则它的幸福度为 R_i 现在给出每只骆驼的 K_i, L_i, R_i 你需要对这些骆驼进行排序，使得 N 只骆驼的幸福度之和最大。
- 题解：考虑每只骆驼的 L_i 与 R_i
 - 若 $L_i = R_i$ 该骆驼位置与结果无关，直接将其 L 加入总幸福中
 - 若 $L_i > R_i$ 记为第一类
 - 若 $L_i < R_i$ ，记为第二类，令其 $K_i = N - K_i$ ，并 $\text{swap}(L_i, R_i)$ 对于第一类，如果排序后在前 i 个，收益为 L ，否则为 R 第二类与第一类相反，如果是后 i 个，收益为 L ，否则为 R 能获取的总收益为两者之和 现在进行排序，比较不同的两只骆驼 i,j ，为使得 $L_i + R_j$

$L_j + R_i$ 即 $L_i - R_i > L_j - R_j$, 满足条件的在前。为使收益最大
记 $f_1, f_2, \dots, f_n, f_i$ 表示前 i 个位置可以插入的数量 即加入一个骆驼 j 令其加入 k_j 个位置, 并令其后面的 f_u 的值减一。如果有 $f_u = 0$, 那么 u 前面已满, 对于 $k_j \leq u$ 的骆驼 j 不能插入前 u 个位置了。

F Two Snake

- 题意：给出一个数 N 现需要选满足下列要求的10个数 $s_1, s_2, n_1, n_2, u_1, u_2, k_1, k_2, e_1, e_2$, 求满足条件的10元组带入式子 $(s_2 - s_1)(n_2 - n_1)(u_2 - u_1)(k_2 - k_1)(e_2 - e_1)$ 的所有结果的和
 - $0 \leq s_1 \leq s_2$
 - $0 \leq n_1 \leq n_2$
 - $0 \leq u_1 \leq u_2$
 - $0 \leq k_1 \leq k_2$
 - $0 \leq e_1 \leq e_2$
 - $s_1 + s_2 + n_1 + n_2 + u_1 + u_2 + k_1 + k_2 + e_1 + e_2 \leq N$
- 题解：暴力，打表用 BM 算法能过。还在研究中，学习完相关知识点在来补。。。

总结：

前三题水题做的倒是快，后面的题就感觉乏力了，知识点的熟练度不够，而且还有好多ACM的知识点还不会。。。需要多做题来掌握这些知识点，开始刷题 PS: markdown 转dokuwiki真的好多错。还需要后期手动改，去搜搜康康有没有什么解决办法。

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:atcoder_aising_programming_contest_2020_quantumbolt

Last update: 2020/07/16 14:05