

哈尔滨理工大学 软件与微电子学院

第十届程序设计竞赛

2020年6月14日

A. Race

- 模拟
- 有一种情况：如果小红在到达终点时小明还在休息，那么在模拟结束后得到的时间可能会出现错误，如果小红获胜，直接输出 $L / V2$ 即可

B. Min Value

- 结构体排序 + 映射
- 如果不考虑求 $i + j$ 的值，那么直接按照绝对值大小排序，对于每个 $1 \leq i < n$ 求 $a[i] + a[i+1]$ 的绝对值的最小值即可
- 我们可以想到把数组元素与第一次该元素出现的下标做映射。如果选出两个元素满足最小值，那么就记录它们映射的和；如果两个元素与当前最小值相等，那么就取它们的映射和与当前记录的映射和的最小值。
- 创建结构体记录元素的值和下标，对结构体数组进行排序。如果两个值的绝对值相同，那么就按照下标从小到大排序，否则按绝对值大小排序。排序后从前到后判断即可。
- 特殊情况：当数组元素非负或者非正时，我们只需要输出排序后结构体数组的前两个元素的值的绝对值之和，以及它们的下标和

C. Coronavirus

- bfs
- 搜索时需要判断这一点以及它的八个方向是否存在*，如果存在*，那么该点不可到达

D. Array

- 思维
- 根据异或的性质，可知 $x \leq y$ 并且 x 和 y 的奇偶性相同，不满足这两个条件输出 -1
- 当 $x == y$ 时，数组中至少存在 1 个元素，为 x
- 其他情况：令 $t = (y - x) / 2$ ，如果 $x \& t == 0$ ，那么需要两个元素 $[x + t, t]$ ；否则至少需要三个元素 $[x, t, t]$

E. Prize

- **Shift-And + 贪心**
- 把所有匹配到的中奖位置存在数组里，因为字符兑奖后就不能再使用，再进行贪心即可得到答案

F. Animal Protection

- dp + 单调栈优化
- 题目n和m的范围比较大，dp的时间复杂度为 $O(n \times m^2)$ 。我们在状态转移时可以使用单调栈来优化，时间复杂度为 $O(n \times m)$

G. XOR

- 思维
- 容易想到我们需要找到N的二进制的最高位，并把最高位后面都变为1即为答案
- 特殊情况：当 $N == 1$ ，结果为0

H. Maze

- dfs + 记忆化
- 在搜索时每搜到一个点就把它加到队列里。搜到尽头时，我们把队列里每个元素的答案都设置为搜到尽头时队列里元素的个数即可

I. Prime

- 线性筛 + 前缀和
- 通过线性筛，得到isprime数组，如果isprime[i]为真，说明i为质数。
- 接下来创建前缀和数组num[i]，代表从0到i质数的个数。对于每次询问，只需要输出num[b] - num[a - 1]即可

J. Compare

- 用字符数组读入a和b，if else 判断即可

K. Walk

- 预处理 + 逆元
- 从 $(1, 1)$ 到 (n, m) 的最短路为向下走 $n - 1$ 步，向右走 $m - 1$ 步，结果为 $C(n + m - 2, n - 1)$
- 求组合数，我们需要求阶乘和阶乘的逆元。由于 mod 为质数，我们只需要用费马小定理即可求得逆元
- 设 $\text{fact}[i]$ 为 i 的阶乘， $\text{ifact}[i]$ 为 i 的阶乘的逆元，那么答案为 $\text{fact}[n + m - 2] * \text{ifact}[m - 1] * \text{ifact}[n - 1] \% \text{mod}$

L. Defeat the monster

- 排序 + 双指针
- 我们先对数组进行排序，创建两个指针*i*和*j*，初始都指向第一个元素的位置。*i*每次向后移动一位，*j*每次向后找到最后一个 $a[j] - a[i] \leq 5$ 的位置，如果*j*到了最后一个元素的位置，则跳出循环。最后只需要输出每轮移动后 $j - i + 1$ 的最大值即可

End