

# Update on Wiki

- 创建了本周训练周报
- 创建了暑期牛客第五次集训界面
- 创建了暑期牛客第六次集训界面

## 团队训练

[2020牛客暑期多校训练营（第五场）](#)

[2020牛客暑期多校训练营（第六场）](#)

## 每周推荐

fyh:牛客第五场A题

题目大意：有一个  $n$  个点  $m$  条边的带权图，你一开始在  $1$  号点，你要按顺序完成  $k$  个任务，第  $i$  个任务是先去  $a_i$  再走到  $b_i$ 。当你走到一个点上的时候，你可以在这个点创建一个传送门。当同时存在两个传送门的时候，你可以在传送门之间不耗代价地传送。如果已经存在了两个传送门，你想再创建一个，就必须选择之前的一个传送门关掉（关掉这个操作不耗时间，并且是远程操作，不需要走过去）。问完成所有任务的最短总行走距离。

tag: 最短路 DP DP状态优化

做法:先考虑最暴力的 DP  $f[i][u][a][b]$  表示当前已经完成了前  $i$  个结点，当前在  $u$  俩传送门分别在  $a$  和  $b$  的最小距离，这个玩意得通过 Dijkstra 转移。然后发现我们没有必要把俩传送门位置都记录，因为剩下一个我们就扔在脚底下就能完全包含，还得通过 Dijkstra 转移。继续想，直接把  $u$  扔掉，当前已经完成了前  $i$  个节点，传送门在  $a$ ，那么我们考虑从  $i$  走到  $i+1$  有几种方式呢？我们可以直接走过去，还可以在脚下丢个传送门再瞬移到另一个传送门再去  $i+1$ 。还可以考虑转移到  $i+1$  的时候传送门到  $b$  了，可以从  $i$  瞬移到  $a$ ，然后走到  $b$ ，在  $b$  处放下传送门，最后再走到  $i+1$ ，还可以直接从  $i$  走到  $b$  放下传送门，然后再瞬移到  $a$  再去  $i+1$  或者直接走去  $i+1$ 。复杂度  $O(2k * N^2)$

comment:考虑优化状态的时候要想到什么是真正有用的

wxg: Educational Codeforces Round 92 (Rated for Div. 2) ECalendar Ambiguity

题目大意：一年有  $m$  个月，每个月  $d$  天，一周是  $k$  天，问你有多少点对  $(a, b)$  满足  $a$  月  $b$  日和  $b$  月  $a$  日在按星期算是同一天

tag: 数学

做法:

comment:

hxm:M-SOLUTIONS Programming Contest 2020 F题

题目大意：二维平面有若干架飞机 ( $n \leq 10^5$ ) 以相同速度分别沿四个方向中的一个方向飞行，问最早什么时候会有飞机相撞？

tag:转化，二分

做法:相向而行的飞机相撞很容易判断，主要讨论交叉相撞的。以向上和向右的飞机为例，如果两个飞机在某个位置相撞那么其实可以把这个相撞的位置上移到一个特定的直线上，同时将向右飞行的飞机向左平移

等同于其距离该直线的距离，这样转化了，两者还是会相撞，但相撞点移到了这条直线。通过这种方法，我们可以把向右移动的飞机都移动到一条直线上，把将问题降维了，对于每个向上的飞机只需在上面二分查找即可。其它方向交叉类似。

comment:问题降维

---

## 个人训练

### 傅云濠

补题：牛客第五场ABK 牛客第六场A

比赛：[Codeforces Round #660 \(Div. 2\)](#)

比赛：[Educational Codeforces Round 92 \(Rated for Div. 2\)](#)

比赛：M-SOLUTIONS Programming Contest 2020

整理板子：

---

### 王兴罡

---

### 黄旭民

补题：牛客第五场C M-SOLUTIONS Programming Contest 2020 F题

比赛：M-SOLUTIONS Programming Contest 2020

整理板子：

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

Permanent link:  
[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:die\\_java:weeksummary8&rev=1596184765](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:die_java:weeksummary8&rev=1596184765)

Last update: 2020/07/31 16:39

