NOIP 模拟赛

2018

tkandi

题目名称	naive 的瓶子	naive 的图	naive 的游戏
目录	colour	graph	game
可执行文件名	colour	graph	game
输入文件名	colour.in	graph.in	game.in
输出文件名	colour.out	graph.out	game.out
每个测试点时间限制	2s	2s	5s
每个测试点空间限制	512MB	512MB	512MB
测试点数目	20	20	20
每个测试点分值	5	5	5
是否有部分分	否	否	否
题目类型	传统型	传统型	传统型
是否有附加文件	是	是	是

提交源程序文件名

对于 C++ 语言	colour.cpp	graph.cpp	game.cpp
对于 C 语言	colour.c	graph.c	game.c
对于 Pascal 语言	colour.pas	graph.pas	game.pas

编译选项

对于 C++ 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 C 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 Pascal 语言	-O2	-O2	-O2

1 naive 的瓶子 (colour)

1.1 题面描述

众所周知，小 naive 有 n 个瓶子，它们在桌子上排成一排。第 i 个瓶子的颜色为 c_i，每个瓶子都有灵性，每次操作可以选择两个**相邻**的瓶子，消耗他们颜色的数值乘积的代价将其中一个瓶子的颜色变成另一个瓶子的颜色。

现在 naive 要让所以瓶子的颜色都一样，操作次数不限，但要使得操作的总代价最小。

1.2 输入格式

输入文件为 $colour.in$。

一个测试点内多组数据。

第一行，一个正整数 T，表示数据组数。

每组数据内：

第一行一个整数 n，为瓶子的个数。

第二行共 n 个整数，第 i 个整数为第 i 个瓶子的颜色 c_i。

1.3 输出格式

输入文件为 $colour.out$。

共 T 行，每行一个整数，为最小的总代价。

1.4 样例输入

1
4
7 4 6 10

1.5 样例输出

92

1.6 样例解释

$\{7\ 4\ 6\ 10\}->\{4\ 4\ 6\ 10\}->\{4\ 4\ 4\ 10\}->\{4\ 4\ 4\ 4\}$。

总代价为 $7\times4+4\times6+4\times10=92$。

1.7 数据范围与约定

$1\leq T\leq10$。

对于测试点内的每组数据：

<table>
<tr><td>测试点编号</td><td>n</td><td>c_i</td><td>特殊性质</td></tr>
<tr><td>1, 2, 3, 4</td><td>≤ 3</td><td>$\leq 10^5$</td><td rowspan="5">数据保证随机生成</td></tr>
<tr><td>5, 6, 7, 8</td><td>≤ 10</td><td rowspan="2">≤ 4</td></tr>
<tr><td>9, 10</td><td rowspan="4">≤ 300</td></tr>
<tr><td>11, 12</td><td>≤ 5</td></tr>
<tr><td>13 - 16</td><td rowspan="2">$\leq 10^5$</td></tr>
<tr><td>17 - 20</td><td>无</td></tr>
</table>

数据保证：$1\leq n\leq300$，$1\leq c_i\leq10^5$。

~~由于出 (ban) 题人太菜，不知道怎么造数据。~~

2　naive 的图 (graph)

2.1　题面描述

众所周知，小 naive 有一张 n 个点，m 条边的带权无向图。第 i 个点的颜色为 c_i。$d(s,t)$ 表示从点 s 到点 t 的权值最小的路径的权值，一条路径的权值定义为路径上权值最大的边的权值。

求所有满足 $u < v, |c_u - c_v| \geq L$ 的点对 (u,v) 的 $d(u,v)$ 之和。

2.2　输入格式

输入文件为 *graph.in*。

第一行，三个整数 n,m,L，表示点数，边数和参数 L。

第二行，n 个整数，第 i 个数为第 i 个点的颜色 c_i。接下来 m 行，每行三个整数 u_i,v_i,w_i，描述了一条边。

2.3　输出格式

输出文件为 *graph.out*。

共一行，一个整数，表示答案。

2.4　样例输入

```
4 5 2
6 4 5 2
1 2 8
2 3 7
2 4 8
1 3 2
1 4 1
```

2.5　样例输出

```
17
```

2.6 样例解释

满足条件的点对：$(1,2),(1,4),(2,4),(3,4)$，答案为 $7+1+7+2=17$。

2.7 数据范围与约定

对于每个测试点内的数据：

测试点编号	n	m	特殊性质
1, 2	≤ 10	≤ 20	无
3 - 6	≤ 1000	≤ 2000	
7 - 10	$\le 2\times 10^5$	$\le 5\times 10^5$	$L=0$
11 - 14			$c_i \le 50$
15 - 16			数据保证随机生成
17 - 20			无

数据保证：$1\le n\le 2\times 10^5$，$1\le m\le 5\times 10^5$，$0\le c_i, L\le 10^9$，$1\le u,v\le n$，$0\le w\le 10^8$，图保证联通。

注意：可能会有重边和自环。

3 naive 的游戏 (game)

3.1 题面描述

众所周知，小 naive 有一个游戏。游戏地图是一个无限的一维数轴，游戏的目标是从起点 s 到终点 t，除了起点其他每个点上都有小怪。有 n 条线段，每条线段上的相邻点之间有桥相连。

他有两种操作：1. 走到相邻的有桥相连的点上，因为上面有小怪，所以要消耗 1 的代价；2. 使用技能跳一跳，跳到距离为 L 的点上，由于把该点上的小怪踩死了，所以不消耗代价。

这个游戏有两种模式，简单和困难。

简单模式：任意点都为可停留点。

困难模式：只有在线段上出现过的点为可停留点。

不可停留点不可到达。

现求从起点到终点所消耗的最小代价。

3.2 输入格式

输入文件为 $game.in$。

一个测试点内多组数据。

第一行，一个正整数 T，表示数据组数。

每组数据内：

第一行，五个整数 $type, n, L, s, t$，分别表示游戏模式，线段数，跳一跳的距离，起点坐标和终点坐标。$type = 0$ 表示为简单模式，$type = 1$ 表示为困难模式。

接下来 n 行，每行两个正整数 l_i, r_i，表示线段 $[l_i, r_i]$。

3.3 输出格式

输出文件为 $game.out$。

共 T 行，每行为该组数据的答案。若不可达输出 -1。

3.4 样例输入

```
4
0 3 4 1 10
1 2
5 6
```

7 10
1 4 5 2 20
1 2
7 8
13 14
19 20
1 3 20 1 19
1 5
10 13
19 20
0 4 20 1 5
1 4
10 12
15 15
17 20

3.5 样例输出

1
3
-1
-1

3.6 样例解释

第一个数据：最优方案之一为 1 => 5 => 9 -> 1，代价为 0 + 0 + 1 = 1。

第二个数据：最优方案之一为 2 => 7 -> 8 => 13 -> 14 => 19 -> 20，代价为 0 + 1 + 0 + 1 + 0 + 1 = 3。

第三个数据：从起点不可到达终点。

第四个数据：从起点不可到达终点。

（用 => 表示使用技能跳一跳，-> 表示直接走）。

3.7 数据范围与约定

$1 \le T \le 10$。

对于测试点内的每组数据：

<table>
<tr><td>测试点编号</td><td>n</td><td>L, l_i, r_i</td><td>特殊性质</td></tr>
<tr><td>1, 2</td><td>≤ 100</td><td rowspan="2">$\le 2 \times 10^6$</td><td rowspan="3">无</td></tr>
<tr><td>3, 4</td><td rowspan="2">≤ 1000</td></tr>
<tr><td>5, 6</td><td>$\le 2 \times 10^9$</td></tr>
<tr><td>7</td><td rowspan="5">$\le 10^5$</td><td rowspan="2">$\le 2 \times 10^6$</td><td>$type = 0$</td></tr>
<tr><td>8</td><td>无</td></tr>
<tr><td>9, 10</td><td rowspan="3">$\le 2 \times 10^9$</td><td>$type = 0$</td></tr>
<tr><td>11 - 14</td><td>数据保证随机生成</td></tr>
<tr><td>15 - 20</td><td>无</td></tr>
</table>

数据保证起点终点都是可停留的点，$l_i \le r_i < l_{i+1}$。

~~由于出 (ban) 题人太菜，造不出强的数据。~~