DP #1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 题目名称 | 匹配 | 块 | 路径 | 染色 |
| 输入文件名 | match.in | block.in | path.in | paint.in |
| 输出文件名 | match.out | block.out | path.out | paint.out |
| 每个测试点时限 | 1s | 1s | 1s | 1.5 s |
| 测试点数目 | 10 | 10 | 10 | 10 |
| 每个测试点分值 | 10 | 10 | 10 | 10 |
| 内存限制 | 256M | 256M | 256M | 256M |
| 是否有部分分 | 否 | 否 | 否 | 否 |
| 题目类型 | 传统 | 传统 | 传统 | 传统 |

注意：代码长度限制均为64K，不开O2。

1 **匹配**(match.c/cpp/pas)

1.1 **题目描述**

现在有一排花和一排花盆，每株花有互不相同的标号，花盆也有互不相同的标号，标号均为1~n，也就是有 n 株花和 n 个花盆。

花与花盆共排成了上下两排，现在希望为尽可能多的花匹配一个对应的花盆，匹配时将为其连上一条直线。要求匹配所产生的直线互不相交，并且为了和谐，匹配的花与花盆标号应当相同。

问最多能匹配的对数。

1.2 **输入格式**

第一行为一个整数n ，分别表示 A、B 的序列长度。

接下来一行 n 个数，表示序列 A。

接下来一行 n 个数，表示序列 B。

1.3 **输出格式**

输出一行一个整数，表示最多能匹配的对数。

1.4 **样例输入**

3

1 2 3

1 3 2

1.5 **样例输出**

2

1.6 **数据范围与约定**

对于20％的数据，n <= 300。

对于40％的数据，n <= 1000。

对于100％的数据，n <= 100000。

2 **块** (block.c/cpp/pas)

2.1 **题目描述**

有一棵n个结点的树，结点从1到n标号。现在希望删去最少的边，使得出现一个大小为 K 的联通块。

2.2 **输入格式**

第一行两个整数n, K，代表树的结点个数，以及希望出现大小为K的联通块。

接下来n-1行，每行两个数 u，v，表示u、v之间有一条连边。

2.3 **输出格式**

输出一行一个数，表示最少删去的边数。

2.4 **样例输入**

11 6

1 2

1 3

1 4

1 5

2 6

2 7

2 8

4 9

4 10

4 11

2.5 **样例输出**

2

2.6 **数据范围与约定**

对于20％的数据，保证1 <= n <= 18。

对于50％的数据，保证1 <= n <= 200。

对于100％的数据，保证1 <= n <= 1000，1<=K<=n。

3 **路径**(path.c/cpp/pas)

3.1 **题目描述**

给出一个有n 个点m 条边的有向图，每个结点上有一个小写字母，定义一条路径的权值是这条路径上出现次数最多的字母的个数，求出这个图最大权值的路径的权值。

3.2 **输入格式**

第一行两个整数n, m，表示结点数和边数。

第二行一个不带空格的字符串，第 i 个字符表示第 i 个结点对应的小写字母。

3.3 **输出格式**

输出一行一个整数，表示最大权值的路径的权值，若为无穷大则输出-1 。

3.4 **样例输入**

5 4

abaca

1 2

1 3

3 4

4 5

3.5 **样例输出**

3

3 .6 **数据范围与约定**

对于20％的数据，保证1 <= n, m <= 10。

对于50％的数据，保证1<=n, m<=5000。

对于100％的数据，保证1<=n, m<=300000。

4 **染色**(paint.c/cpp/pas)

4.1 **题目描述**

一个木板上有 n 个格子，格子一开始均无色（颜色为 0）。现在要进行 K 次染色，每次告诉你染区间 [li,ri]，颜色为 ci。颜色会被覆盖。问最后的木板上所有格子的颜色。

4.2 **输入格式**

第一行两个整数 n; K，表示格子数和染色次数。

接下来 K 行，每行三个整数 li，ri，ci，表示染色区间和颜色

4.3 **输出格式**

输出一行 n 个整数，输出这个木板上的所有格子颜色。

4.4 **样例输入**

4 2

1 2 4

2 4 1

4.5 **样例输出**

4 1 1 1

4.6 **数据范围与约定**

对于10％的数据，保证 n <= 100。

对于40％的数据，保证n, K<=100000。

对于100％的数据，保证1<=n, K<=1000000，1<=ci<=1000000。