

准考证号：SH-B260101

(请选手务必仔细阅读本页内容)

一、题目概况

中文题目名称	人员分配	高频词汇	飞盘	仪仗队	车站分级
英文题目与子目录名	assign	frequency	frisbee	parade	level
可执行文件名	assign	frequency	frisbee	parade	level
输入文件名	assign.in	frequency.in	frisbee.in	parade.in	level.in
输出文件名	assign.out	frequency.out	frisbee.out	parade.out	level.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒	1 秒
测试点数目	10	10	10	10	10
每个测试点分值	10	10	10	10	10
附加样例文件	无	无	无	无	无
结果比较方式	全文比较，过滤末行后空行，不过滤中间行行末空格				
题目类型	传统	传统	传统	传统	传统
运行内存上限	512M	512M	512M	512M	512M

二、提交源程序文件名

对于 C++ 语言	assign.cpp	frequency .cpp	frisbee .cpp	parade.cpp	level.cpp
-----------	------------	----------------	--------------	------------	-----------

三、编译命令 (不包含任何优化开关)

对于 C++ 语言	<code>g++ -o assign - 1m</code>	<code>g++ -o frequency frequency - 1m</code>	<code>g++ -o frisbee frisbee.cpp - 1m</code>	<code>g++ -o frequency parade.cpp - 1m</code>	<code>g++ -o level level .cpp -1m</code>
-----------	---------------------------------	--	--	---	--

人员分配

(assign.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

学校体育社团最近新招收了 $2N$ 名社员，你作为社长需要复制将这 $2N$ 名社员分配到篮球队和足球队各 N 名。

于是你调查了各新社员对于篮球队和足球队的意愿，我们用 B_i 和 F_i 表示。 B_i 代表如果分配第 i 名社员去篮球队，他的满意指数； F_i 代表如果分配第 i 名社员去足球队，他的满意指数。

请问如何分配才能使 $2N$ 名社员的满意指数之和最高。

【输入格式】

输入文件 assign.in

输入第一行，包含一个整数 N 。

之后是 $2N$ 行，每行包含两个整数 B_i 和 F_i

【输出格式】

输出文件 assign.out

输出一个整数表示。

【输入输出样例 1】

factor.in	factor.out
2	310
100 50	
80 80	
50 100	
10 30	

【说明】前两个人去篮球队($100+80=180$)，后两个人去足球队($100+30=130$)，总数为 $180+130=310$ 。

【数据规模与约定】

对于 20%的数据， $N \leq 10$;

对于 100%的数据， $N \leq 1000$ ， $0 \leq B_i, F_i \leq 10000$ 。

高频词汇

(frequency.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

张三在学习英语时最害怕背单词，为此他准备编写一个程序：统计一下最近五年中考试卷中单词出现的频率，然后按照频率从高到底的顺序重点背前 K 个。也就是说：当两个单词出现的次数相同时，则按照字典序排序，否则按单词出现的次数从高到底排序。

【输入格式】 输入文件 frequency.in

输入有两行，输入第一行为一段英语文章，文章中有多少个单词无法提前知晓，但可以保证所有的单词都是小写字母，由空格隔开，且单词长度不超过 20，单词的个数不超过 500。输入的第二行为一个正整数 K ， K 的取值范围为最小为 1，最大不超过不同单词的数量。

【输出格式】 输出文件 frequency.out

输出共 K 行，每行输出一个单词，以出现次数高低排序，次数相同时按字典序排序。

【输入输出样例 1】

frequency.in	frequency.out
i love tiger i love coding 2	i love

【说明】 i 和 $love$ 为出现次数最多的两个单词，均为 2 次。按字母顺序 i 在 $love$ 之前。

【输入输出样例 2】

frequency.in	frequency.out
today is history today is 3	today is history

【说明】 $today$ 、 is 、 $history$ 是出现次数最多的三个单词，出现次数依次为 3、2、1。

【数据规模与约定】

对于 100% 的数据： $1 \leq \text{单词个数} \leq 500$ ； $1 \leq \text{单词长度} \leq 20$ ；英文文章只包含小写字母； K 的取值范围为 $[1, \text{不同单词数量}]$ 。

飞盘

(frisbee.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

N 名哨兵的高度为 $1, 2, \dots, N$ 。一天，哨兵以某个顺序排成一行玩飞盘；令 $h_1 \dots h_N$ 表示此顺序下哨兵们的高度（因此 h 是 $1 \dots N$ 的一个排列）。

队伍中位于位置 i 和 j 的两名哨兵可以成功地来回扔飞盘当且仅当他们之间的每名哨兵的高度都低于 $\min(h_i, h_j)$ 。请计算所有可以成功地来回扔飞盘的哨兵所在的位置对 $i < j$ 之间的距离总和。位置 i 和 j 之间的距离为 $j - i + 1$ 。

【输入格式】

输入文件 frisbee.in

输入的第一行包含一个整数 N 。第二行包含 $h_1 \dots h_N$ ，用空格分隔。

【输出格式】

输出文件 frisbee.out

输出可以成功地来回扔飞盘的哨兵所在的位置对 $i < j$ 之间的距离总和。

【输入输出样例 1】

frisbee.in	frisbee.out
7 4 3 1 2 5 6 7	24

【说明】

这个例子中可以成功的位置对如下：

(1, 2), (1, 5), (2, 3), (2, 4), (2, 5), (3, 4), (4, 5), (5, 6), (6, 7)

【数据说明】

对于 30% 的数据， $N \leq 5000$

对于 100% 的数据， $N \leq 300000$ $h_i \leq 1000000000$

仪仗队 (parade)

(parade.cpp)

时空限制：1s/512M，测试数据共 10 组

【题目描述】

已知西佳佳大学共有 k 个不同的班级，第 i 个班级参与到开幕式入场仪式的人数为正整数 c_i ，这 c_i 个人的身高数字信息均为已知信息。你需要从每个班级里选出恰好 1 位代表组成学校仪仗队，第 i 个班级的代表身高记作 h_i 。由此，可以确定仪仗队的身高集合 $H = \{h_1, h_2, \dots, h_k\}$ 。集合 H 内允许重复数字出现。请问，集合 H 的极差最小是几？

备注：一个数字集合的极差就是该集合中的最大值减去最小值。

【输入格式】 从文件 parade.in 中读入数据。

输入的第一行包含一个正整数 k ，表示班级数， $1 \leq k \leq 10^5$ 。接下去共 k 行，每行表示一个班级的情况。关于第 i 个班级的情况，首先输入一个正整数 c_i ，然后输入 c_i 个正整数，表示 c_i 个人的身高，个人身高均不超过 10^9 。保证：所有 c_i 总和不超过 10^6 。

【输出格式】 输出到文件 parade.out 中。

输出一行，包含 1 个整数。

【输入输出样例 1】

parade.in	parade.out
3 5 9 4 6 7 8 3 1 5 6 2 10 3	2

【说明】 第一个班级选 4，第二个班级选 5，第三个班级选 3。

【样例 2】 见目录下 parade2.in 与 parade2.ans。

【数据范围】

测试点编号	特殊性质
1	$k=2$
2	所有人的身高均不超过 2
3	所有 c_i 均为 1
4	所有 c_i 均为 2
5~10	无

车站分级

(level.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

一条单向的铁路上，依次有编号为 1, 2, ..., n 的 n 个火车站。每个火车站都有一个级别，最低为 1 级。现有若干趟车次在这条线路上行驶，每一趟都满足如下要求：如果这趟车次停靠了火车站 x，则始发站、终点站之间所有级别大于等于火车站 x 的都必须停靠。（注意：起始站和终点站自然也算作事先已知需要停靠的站点）例如，下表是 5 趟车次的运行情况。其中，前 4 趟车次均满足要求，而第 5 趟车次由于停靠了 3 号火车站（2 级）却未停靠途经的 6 号火车站（亦为 2 级）而不满足要求。现有 m 趟车次的运行情况（全部满足要求），试推算这 n 个火车站至少分为几个不同的级别。

车站编号	1	2	3	4	5	6	7	8	9
车站级别	3	1	2	1	3	2	1	1	3
车次									
1	始	→	→	停	→	→	→	停	→
2	始	→	→	→	停	→	终		
3	始	→	→	→	→	→	→	→	→
4	始	→	→	→	→	→	→	→	→
5	始	→	→	→	→	→	→	→	→

【输入格式】

输入文件 level.in 第一行包含 2 个正整数 n, m，用一个空格隔开。

第 i + 1 行 (1 ≤ i ≤ m) 中，首先是一个正整数 si (2 ≤ si ≤ n)，表示第 i 趟车次有 si 个停靠站；接下来有 si 个正整数，表示所有停靠站的编号，从小到大排列。每两个数之间用一个空格隔开。输入保证所有的车次都满足要求。

【输出格式】

输出文件 level.out 输出只有一行，包含一个正整数，即 n 个火车站最少划分的级别数。

【输入输出样例 1】

level.in	level.out
9 2 4 1 3 5 6 3 3 5 6	2

【说明】：由第一趟车能得出：1、3、5、6 可以为同一个级别，2、4 低一个级别；由第二趟车能得出：3、5、6 可以为同一级别，2、4 低一个级别。综合两趟车的结论，1-6 号车站至少要划分为 2 个级别。

【输入输出样例 2】

level.in	level.out
9 3	3
4 1 3 5 6	
3 3 5 6	
3 1 5 9	

【数据规模】

对于 20%的数据, $1 \leq n, m \leq 10$;

对于 50%的数据, $1 \leq n, m \leq 100$;

对于 100%的数据, $1 \leq n, m \leq 1000$ 。