CCF 全国信息学奥林匹克联赛（NOIP2014）复赛

提高组 day2

（请选手务必仔细阅读本页内容）

一．题目概况

中文题目名称	第一题	整数划分	观光旅行
英文题目与子目录名	eat	division	trip
可执行文件名	eat	division	trip
输入文件名	eat.in	division.in	trip.in
输出文件名	eat.out	division.out	trip.out
每个测试点时限	1 秒	1 秒	1 秒
测试点数目	10	20	20
每个测试点分值	10	5	5
附加样例文件	有	有	有
结果比较方式	全文比较（过滤行末空格及文末回车）		
题目类型	传统	传统	传统
运行内存上限	128M	128M	128M

二．提交源程序文件名

对于 C++语言	eat.cpp	division.cpp	trip.cpp
对于 C 语言	eat.c	division.c	trip.c
对于 pascal 语言	eat.pas	division.pas	trip.pas

三．编译命令（不包含任何优化开关）

对于 C++语言	g++ -o eat eat.cpp -lm	g++ -o division division.cpp -lm	g++ -o trip trip.cpp –lm
对于 C 语言	gcc -o eat eat.c -lm	gcc -o division division.c -lm	gcc -o trip trip.c -lm
对于 pascal 语言	fpc eat.pas	fpc division.pas	fpc trip.pas

注意事项：

1、 文件名（程序名和输入输出文件名）必须使用英文小写。
2、 C/C++中函数 main()的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3、 全国统一评测时采用的机器配置为：CPU AMD Athlon(tm) 64x2 Dual Core CPU 5200+，2.71GHz，内存 2G，上述时限以此配置为准。
4、 只提供 Linux 格式附加样例文件。
5、 特别提醒：评测在 NOI Linux 下进行。

1.第一题

(eat.cpp/c/pas)

【问题描述】

凶猛的**出来吃人了！

每天早晨，**从大山里出来，到达一个城市，然后花费一整天的时间把这座城市里的人吃光。直到夜晚，**才回到山中去。当**经过一个城市时，不管是否吃人，它都会把这座城市彻底破坏，以至于下次不能再到这个城市吃人了。

显然，城市里的居民无法忍受这样的状况。所以，每天夜晚，每座城市里都会有一个人逃到乡下去，到了乡下以后**就永远吃不到他了。

城市之间有一些双向道路连接着。其中 1 号城市连接着大山，即**每天的旅途的起点。当然，**只能沿着这些道路走。

意识到必须抓紧时间吃人，所以它每天都要认真选取要去的城市，当然它不能选择已经被吃过或破坏过的城市。现在问题来了，在所有城市没有人居住之前，最多能吃掉多少人？

【输入】

输入文件名为 eat.in。

第一行两个整数n, m，用一个空格隔开，表示城市的个数和道路数。

第二行n个整数a_i，表示每座城市初始的人数。两个数之间用一个空格隔开。

接下来m行，每行两个整数$u, v(1 \le u, v \le n, u \ne v)$，用一个空格隔开，表示城市$u$和城市$v$之间有一条双向道路。城市 1 和大山之间也有一条双向道路。

数据保证所有城市都存在到城市 1 的路径。

【输出】

输出文件名为 eat.out。

输出共一行一个整数，表示在所有城市没有人居住之前，**最多能吃掉的人数。

【输入输出样例】

eat.in	eat.out
5 5 1 3 2 4 7 1 2 1 3 2 3 2 4 3 5	11

【数据说明】

对于 10%的数据，$1 \le n \le 5$，$0 \le m \le 10$，$0 \le a_i \le 5$。

对于 30%的数据，$1 \le n \le 200$，$0 \le m \le 500$，$0 \le a_i \le 200$。

对于 60%的数据，$1 \le n \le 2{,}000$，$0 \le m \le 10{,}000$，$0 \le a_i \le 20{,}000$。

对于 100%的数据，$1 \le n \le 200{,}000$，$0 \le m \le 2{,}000{,}000$，$0 \le a_i \le 2{,}000{,}000$。

2.整数划分

(division.cpp/c/pas)

【问题描述】

BG 得到了一个整数N，他想要把N分解成若干个小整数的乘积。

BG 给出了他的分解规则：

- 分解出的整数必须来自集合S；
- 分解出的整数必须互不相同，且两两互质。

现在给出整数N，集合大小M和集合S，求 BG 有多少种分解方法。

【输入】

输入文件名为 division.in。

第一行两个整数N, M，表示要分解的数和集合大小。

第二行包含M个互不相同的整数a_i，描述了集合S，即$S = \{a_1, a_2, a_3, \dots, a_M\}$。

【输出】

输出文件名为 division.out。

输出共一行一个整数，表示方案数。数据保证答案在 64 位带符号整数范围内。

如果没有方案，输出一个 0。

【输入输出样例 1】

division.in	division.out
12 5 2 3 4 5 6	1

【输入输出样例说明】

共 1 种方案：3×4。

【输入输出样例 2】

division.in	division.out
42 8 1 2 3 6 7 14 21 42	10

【输入输出样例说明】

共 10 种方案：42，1×42，2×21，3×14，6×7，$1\times2\times21$，$1\times3\times14$，$1\times6\times7$，$2\times3\times7$和$1\times2\times3\times7$。

【数据说明】

对于 10%的数据，$2\le N\le20$，$1\le M\le5$，$1\le a_i\le10$；

对于 20%的数据，$2\le N\le1{,}000$，$1\le M\le20$，$1\le a_i\le100$；

对于 40%的数据，$2\le N\le1{,}000{,}000$，$1\le M\le80$，$1\le a_i\le10{,}000$；

对于 70%的数据，$2\le N\le1{,}000{,}000{,}000$，$1\le M\le200$，$1\le a_i\le10{,}000{,}000$；

对于 100%的数据，$2\le N\le10^{18}$，$1\le M\le500$，$1\le a_i\le1{,}000{,}000{,}000$。

3.观光旅行

(trip.cpp/c/pas)

【问题描述】

BG 来到了一个美丽的风景区旅行，这个景区共有n个景点，编号从1到n。这n个景点之间共有m条双向的观光道路，每条观光道路都有一个魅力值，第i条观光道路的魅力值为w_i。

现在 BG 想从任意一个景点出发，沿着一条路径旅行其它景点。为了避免旅途的枯燥，BG 每次经过的道路的魅力值都要严格大于之前经过的任何道路。同时，他想让旅途尽可能长。

请问在满足他的要求的情况下，路径的最长长度是多少，并求出不同的最长路径共有多少条。路径的长度即它经过的道路数。两条路径被认为不同，当且仅当它们经过的景点序列不同。为了防止输出的数字过大，你只需输出路径数对1,000,000,007取模的结果。

【输入】

输入文件名为 trip.in。

输入文件的第一行有两个用一个空格隔开的整数n、m，表示该景区有n个景点和m条观光道路。

接下来m行每行三个整数u_i,v_i,w_i，每两个整数之间用一个空格隔开，表示第i条道路的魅力值为w_i，在景点u_i和v_i之间。

输入数据保证$u_i\neq v_i$，两个景点之间最多只有一条观光道路，第一问的答案至少为 2。

【输出】

输出文件名为 trip.out。

输出的第一行一个整数，表示最长的路径长度。

输出的第二行一个整数，表示不同的最长路径数对1,000,000,007取模的结果。

【输入输出样例】

trip.in	**trip.out**
5 5 1 2 4 1 3 5 2 4 3 2 3 1 2 5 2	3 3

【输入输出样例说明】

共有 3 条长度为 3 的合法路径：

1. 3—2—1—3，经过道路的魅力值为 1、4、5；
2. 4—2—1—3，经过道路的魅力值为 3、4、5；
3. 5—2—1—3，经过道路的魅力值为 2、4、5。

【数据说明】

对于 10%的数据，$1 \le n \le 10$，$1 \le m \le 20$；

对于 30%的数据，$1 \le n \le 100$，$1 \le m \le 500$；

对于 60%的数据，$1 \le n \le 1{,}000$，$1 \le m \le 3{,}000$；

对于 80%的数据，$1 \le n \le 2{,}000$，$1 \le m \le 100{,}000$；

对于 100%的数据，$1 \le n \le 50{,}000$，$1 \le m \le 200{,}000$，$0 \le w_i \le 10^9$。