

A、平均

内存限制：512 MB 时间限制：1000 ms 文件名：avg.cpp/in/out

题目描述

梦梦在一次考试中考砸了，还没有考到班级的平均分。

这场考试梦梦的班级一共有 n 个人参加，其中第 i 个人的最终得分为 a_i 。

梦梦很好奇，如果只选出其中的一些人（至少一个人）重新组成一个班级，所有人的考试分数不变的情况下，超过新班级平均分的人数最多可以是多少人，请你输出这个值。

输入格式

第一行包含一个正整数，表示 n 。

第二行包含 n 个整数，第 i 个表示 a_i 。

输出格式

输出 1 个整数，表示最终答案。

样例 1 输入

```
1 | 5
2 | 1 2 3 4 5
```

样例 1 输出

```
1 | 2
```

样例 1 解析

一组合法解为选择 (1, 2, 3, 4)，平均数为 2.5，严格大于平均数的个数共 2 个。

样例 2 输入

```
1 | 18
2 | 3 5 5 1 2 2 3 9 2 2 7 1 7 3 2 3 6 6
```

样例 2 输出

```
1 | 7
```

样例 3 输入

```
1 | 20
2 | 170421784 139902913 635488787 31946330 289281525 541340761 264115901
   680775011 102068389 128171044 636359374 477042327 879154459 716982403
   654668380 256334408 874949295 737194459 663371810 549070350
```

样例 3 输出

```
1 | 12
```

数据范围

对于 20% 的数据, $1 \leq n \leq 20$
对于 40% 的数据, $1 \leq n \leq 500$
对于 60% 的数据, $1 \leq n \leq 5000$
对于所有测评数据, $1 \leq n \leq 10^6, 1 \leq a_i \leq 10^9$ 。

B、操作

内存限制：512 MB 时间限制：2000 ms 文件名：opn.cpp/in/out

题目描述

梦梦给出了一个长度为 n 的序列 a , 你可以执行以下操作若干次:

选择一个区间 l, r , 满足 $1 \leq l \leq r \leq n$, 并选定一个整数 k , 将 $a_l, a_{l+1}, a_{l+2}, \dots, a_r$ 的值全部加上 k 。

请问至少执行多少次操作, 可以将 a 序列中所有的值均修改为 0。

输入格式

第一行, 包含一个正整数 n 。
第二行, 包含 n 个整数 a_1, a_2, \dots, a_n 。

输出格式

输出一行, 包含一个整数, 表示答案。

样例 1 输入

```
1 | 5
2 | 1 2 3 2 1
```

样例 1 输出

```
1 | 3
```

样例 2 输入

```
1 | 6
2 | 1 1 4 5 1 4
```

样例 2 输出

```
1 | 4
```

样例 3 输入

```
1 | 23
2 | 500000000 500000000 500000066 500000073 500000073 500000182 500000182
   | 500000087 500000169 500000169 500000133 500000133 500000035 500000035
   | 499999871 499999865 499999865 499999865 500000154 500000072 500000072
   | 500000096 500000000
```

样例 3 输出

```
1 | 11
```

数据范围

对于 20% 的数据, $1 \leq n \leq 2$ 。

对于 40% 的数据, $1 \leq n \leq 5$ 。

对于 70% 的数据, $1 \leq n \leq 15$ 。

对于 100% 的数据, $1 \leq n \leq 23, 0 \leq a_i \leq 10^9$ 。

C、选择排序

内存限制：512 MB 时间限制：3000 ms 文件名：sort.cpp/in/out

题目描述

对于正整数 n 和一个长度为 n 的全排列 p , 考虑以下选择排序程序：

```
1 | void solve(){
2 |     for(int i=1;i<n;i++){
3 |         c[i]=0;
4 |         for(int j=i+1;j<=n;j++){
5 |             if(p[i]>p[j]){
6 |                 swap(p[i],p[j]);
7 |                 c[i]++;
8 |             }
9 |         }
10 |    }
```

根据排序过程, 会得到一个序列 c , 显然 $c[n] = 0$ 。

现给定正整数 n , 以及 $c[1], c[2], \dots, c[n-1]$, 询问有多少个长度为 n 的全排列 p , 调用以上函数后, 所得 c 与输入一致, 答案对 998244353 取模。

输入格式

第一行包含一个正整数 n 。

第二行包含 $n - 1$ 个非负整数 $c[1], c[2], \dots, c[n - 1]$ 。

输出格式

输出一行，包含一个整数，表示答案，答案对 998244353 取模。

样例 1 输入

```
1 | 3
2 | 1 1
```

样例 1 输出

```
1 | 2
```

样例 2 输入

```
1 | 6
2 | 0 1 1 1 0
```

样例 2 输出

```
1 | 4
```

样例 3 输入

```
1 | 6
2 | 0 1 2 1 1
```

样例 3 输出

```
1 | 6
```

数据范围

对于 30% 的数据， $1 \leq n \leq 9$ 。

对于 60% 的数据， $1 \leq n \leq 10^3$ 。

对于 100% 的数据， $1 \leq n \leq 2 \times 10^5, 0 \leq c_i \leq n$ ，保证至少存在一组合法解。

D、树

内存限制：512 MB 时间限制：2000 ms 文件名：tree.cpp/in/out

题目描述

给定一棵 n 个点有根树，其中根节点为 1 号节点。

对于一棵树，有轻重边这一概念，每个点和他的孩子节点之间的边可能是轻边，也可能是重边，初始时所有边均为轻边。

定义 $\text{access}(u)$ 这一函数为依次执行以下两项操作：

1. 对于 1 节点到 u 节点简单路径上的每个点 x ，将至少一端连向 x 的边均更改为轻边。
2. 将 1 节点到 u 节点简单路径上的每条边均修改为重边。

给定一种可能的轻重边状态，询问有多少个长度为 n 的排列，对于一棵初始时全为轻边的树，依次执行 $\text{access}(p_1), \text{access}(p_2), \dots, \text{access}(p_n)$ ，所得的轻重边状态与给定的状态相同。

本题有 q 组询问，我们用如下方式给出每组询问的轻重边状态：

定义 T_0 为所有边均为轻边，给出长度为 q 的序列 x_i ， T_i 为 T_{i-1} 执行 $\text{access}(x_i)$ 后所得的轻重边状态。

你需要对于 T_1, T_2, \dots, T_q 均输出最终答案，答案对 998244353 取模。

输入格式

第一行两个正整数 n, q 。

第二行给出 $n - 1$ 个正整数，第 i 个正整数表示 fa_{i+1} ，表示 $i + 1$ 节点的父亲节点，保证 $fa_{i+1} \leq i$ 。

之后 q 行，每行给出一个正整数 x_i ，满足 $1 \leq x_i \leq n$ 。

输出格式

输出 q 行，每行包含一个整数，表示答案。

样例 1 输入

```
1 8 9
2 1 2 1 2 5 5 2
3 6
4 2
5 6
6 8
7 3
8 1
9 1
10 5
11 8
```

样例 1 输出

```
1 5040
2 1680
3 5040
4 1680
5 1680
6 280
7 280
8 5040
9 1680
```

数据范围

对于 30% 的数据, $1 \leq n, q \leq 1000$ 。

对于另外 30% 的数据, 保证 $\forall 1 \leq i < n, fa_{i+1} = i$ 。

对于 100% 的数据, $1 \leq n, q \leq 10^5$ 。