福建省青少年信息学奥林匹克冬令营

FCS NOI 2018

模拟训练

时间：2018 年 2 月 8 日 12:30 ~ 17:00

题目名称	最大真因数	欧拉函数	残缺的算式
题目类型	传统型	传统型	传统型
目录	`factor`	`phi`	`expr`
可执行文件名	`factor`	`phi`	`expr`
输入文件名	`factor.in`	`phi.in`	`expr.in`
输出文件名	`factor.out`	`phi.out`	`expr.out`
每个测试点时限	2 秒	2 秒	2 秒
内存限制	512 MB	512 MB	512 MB
测试点数目	10	20	25
每个测试点分值	10	5	4

提交源程序文件名

对于 C++ 语言	`factor.cpp`	`phi.cpp`	`expr.cpp`
对于 C 语言	`factor.c`	`phi.c`	`expr.c`
对于 Pascal 语言	`factor.pas`	`phi.pas`	`expr.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`

最大真因数（factor）

【题目描述】

一个合数的真因数是指这个数不包括其本身的所有因数，例如 6 的正因数有 $1,2,3,6$，其中真因数有 $1,2,3$。一个合数的最大真因数则是这个数的所有真因数中最大的一个，例如 6 的最大真因数为 3。

给定正整数 l 和 r，请你求出 l 和 r 之间（包括 l 和 r）所有合数的最大真因数之和。

【输入格式】

从文件 ***factor.in*** 中读入数据。

输入共一行，包含两个正整数 l 和 r。保证 $l \le r$。

【输出格式】

输出到文件 ***factor.out*** 中。

输出共一行，包含一个整数，表示 $[l,r]$ 内所有合数的最大真因数之和。

【样例 1 输入】

```
1 10
```

【样例 1 输出】

```
17
```

【样例 1 解释】

在 1 至 10 之间的合数有 $4,6,8,9,10$，它们的最大真因数分别为 $2,3,4,3,5$，因此最大真因数之和为 $2+3+4+3+5=17$。

【样例 2 输入】

```
101 1000
```

【样例 2 输出】

```
163446
```

【样例 3 输入】

```
180208 975313
```

【样例 3 输出】

```
151642139152
```

【样例 4 输入】

```
339762200 340762189
```

【样例 4 输出】

```
112318862921546
```

【样例 5 输入】

```
2500000000 5000000000
```

【样例 5 输出】

```
3094668961678105770
```

【子任务】

子任务会给出部分测试数据的特点。如果你在解决题目中遇到了困难，可以尝试只解决一部分测试数据。

每个测试点的数据规模及特点如下表：

测试点编号	l, r	约定
1	≤ 100	无
2	≤ 1000	
3	$\le 10^4$	
4	$\le 10^5$	
5	$\le 5 \times 10^6$	
6	$\le 10^7$	
7	$\le 10^9$	$r - l \le 10^6$
8	$\le 5 \times 10^9$	
9	$\le 10^9$	无
10	$\le 5 \times 10^9$	

欧拉函数（phi）

【题目描述】

对于正整数 n，定义欧拉函数 $\varphi(n)$ 为小于等于 n 且与 n 互质的正整数个数。例如 $\varphi(1) = 1$，$\varphi(8) = 4$。

给定正整数序列 $a_1, a_2, \cdots, a_n$，请依次执行 q 个操作，操作有以下三种类型：

- `0 i x`：修改 a_i 的值为 x；
- `1 l r`：查询 $\varphi(a_l + a_{l+1} + \cdots + a_r)$ 的值，输出这个值对 $10^9 + 7$ 取模的结果；
- `2 l r`：查询 $\varphi(a_l \times a_{l+1} \times \cdots \times a_r)$ 的值，输出这个值对 $10^9 + 7$ 取模的结果。

【输入格式】

从文件 ***phi.in*** 中读入数据。

输入的第一行包含两个正整数 n, q，分别表示序列长度及操作个数。

第二行包含 n 个正整数 $a_1, a_2, \cdots, a_n$，表示初始序列。

接下来 q 行，每行三个整数 `0 i x` 或 `1 l r` 或 `2 l r`，表示一个操作。保证 $1 \le i \le n$，$x \ge 1$，$1 \le l \le r \le n$。

【输出格式】

输出到文件 ***phi.out*** 中。

对于每次形如 `1 l r` 或 `2 l r` 操作，输出一行，表示所求的值对 $10^9 + 7$ 取模的结果。

【样例 1 输入】

```
5 10
1 3 5 7 9
1 2 4
0 3 3
1 1 4
2 1 4
0 3 4
2 1 3
0 4 5
1 3 5
1 1 5
2 1 5
```

【样例 1 输出】

```
8
6
36
4
6
10
144
```

【样例 1 解释】

初始序列为 $1,3,5,7,9$，依次进行的 10 个操作如下：

1. 查询 $\varphi(3+5+7)=\varphi(15)=8$，输出 8；
2. 修改 a_3 的值为 3，此时的序列为 $1,3,3,7,9$；
3. 查询 $\varphi(1+3+3+7)=\varphi(14)=6$，输出 6；
4. 查询 $\varphi(1\times3\times3\times7)=\varphi(63)=36$，输出 36；
5. 修改 a_3 的值为 4，此时的序列为 $1,3,4,7,9$；
6. 查询 $\varphi(1\times3\times4)=\varphi(12)=4$，输出 4；
7. 修改 a_4 的值为 5，此时的序列为 $1,3,4,5,9$；
8. 查询 $\varphi(4+5+9)=\varphi(18)=6$，输出 6；
9. 查询 $\varphi(1+3+4+5+9)=\varphi(22)=10$，输出 10；
10. 查询 $\varphi(1\times3\times4\times5\times9)=\varphi(540)=144$，输出 144。

【样例 2】

见选手目录下的 ***phi/phi2.in*** 与 ***phi/phi2.ans***。
该组样例的数据范围同第 9 个测试点。

【样例 3】

见选手目录下的 ***phi/phi3.in*** 与 ***phi/phi3.ans***。
该组样例的数据范围同第 12 个测试点。

【样例 4】

见选手目录下的 ***phi/phi4.in*** 与 ***phi/phi4.ans***。
该组样例的数据范围同第 16 个测试点。

【样例 5】

见选手目录下的 ***phi/phi5.in*** 与 ***phi/phi5.ans***。

该组样例的数据范围同第 20 个测试点。

【子任务】

每个测试点的数据规模及特点如下表：

<table>
<tr><th>测试点编号</th><th>n</th><th>a_i, x</th><th>q</th><th>包含的操作类型</th></tr>
<tr><td>1</td><td rowspan="5">≤ 5</td><td rowspan="5">≤ 10</td><td rowspan="5">≤ 10</td><td rowspan="2">1</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td><td rowspan="2">2</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td><td rowspan="3">0,1,2</td></tr>
<tr><td>6</td><td rowspan="2">≤ 500</td><td rowspan="2">≤ 1000</td><td rowspan="2">≤ 100</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td><td rowspan="5">≤ 5000</td><td rowspan="5">≤ 10000</td><td>≤ 1000</td><td rowspan="2">0,1</td></tr>
<tr><td>9</td><td>≤ 5000</td></tr>
<tr><td>10</td><td>≤ 1000</td><td rowspan="3">0,1,2</td></tr>
<tr><td>11</td><td>≤ 3000</td></tr>
<tr><td>12</td><td>≤ 5000</td></tr>
<tr><td>13</td><td rowspan="8">≤ 50000</td><td rowspan="8">≤ 40000</td><td>≤ 50000</td><td rowspan="4">0,1</td></tr>
<tr><td>14</td><td>≤ 75000</td></tr>
<tr><td>15</td><td rowspan="2">≤ 100000</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td><td>≤ 25000</td><td rowspan="4">0,1,2</td></tr>
<tr><td>18</td><td>≤ 50000</td></tr>
<tr><td>19</td><td>≤ 75000</td></tr>
<tr><td>20</td><td>≤ 100000</td></tr>
</table>

表中“包含的操作类型”表示对应测试点中每个操作的**第一个数**的可能值，例如，包含的操作类型为“0,1”表示该测试点不会出现形如 `2 l r` 的操作。

对于全部测试点，$n \le 50000$，$q \le 100000$，操作 0 的个数不超过 20000，所有的 a_i、操作 0 中的 i, x 及操作 1,2 中的 l, r 均在给定的限制下内均匀随机生成。

残缺的算式（expr）

【题目描述】

小明有一套玩具卡片。这套卡片中，有 n 张是数字卡片，这 n 张数字卡片上分别写有一个正整数 $1, 2, \cdots, n$；还有若干张符号卡片，每张符号卡片上写有符号 +、-、*、(、) 之一。

小明最喜欢用这套卡片来摆算式，然后计算它的结果。当然，摆算式时不必用完所有的卡片。

一天，小明摆好了一个算式。然而，准备计算前，一个熊孩子把算式中的数字卡片全部取走了，只剩下符号卡片和原来数字卡片所在的空位。也就是说，现在的算式可能是这样的：(_+_*_)*(_*_-_)+_（这里 _ 代表原来放有数字卡片的空位）。

无奈小明已不记得原来每个位置填的数字卡片上的数分别是多少，他只能假设每种可能的算式作为原算式的概率相等。现在，小明想知道，原算式的结果的期望值是多少？

形式化地，设有 m 种可能的算式，它们的运算结果分别为 $s_1, s_2, \cdots, s_m$，那么原算式的结果的期望值 E 为

$$E = \frac{1}{m} \sum_{i=1}^{m} s_i$$

【输入格式】

从文件 ***expr.in*** 中读入数据。

第一行一个正整数 n，表示小明拥有的数字卡片的数量。

第二行一个字符串 S，表示数字卡片被取走后的算式。保证 S 中仅包含字符 _、+、-、*、(、)，其中 _ 不超过 n 个，且将所有 _ 当成整数时，S 是一个合法的算术表达式。

【输出格式】

输出到文件 ***expr.out*** 中。

输出一个整数，表示原算式的结果的期望值对 $10^9 + 7$ 取模的结果。

假设所求的答案为 $\frac{a}{b}$（a, b 为整数且互质），定义 $\frac{a}{b}$ 对正整数 p 取模的结果为一个整数 x，满足 $0 \le x < p$ 且 $bx \equiv a \pmod{p}$。可以证明，对于本题的答案及模数，这样的 x 是唯一存在的。

【样例 1 输入】

```
3
_+_
```

【样例 1 输出】

```
4
```

【样例 1 解释】

小明有 3 张数字卡片，分别写有数字 1,2,3，将这 3 张数字卡片填入算式 _+_，可能得到 6 种不同的算式，结果分别如下：

1. $1+2=3$；
2. $1+3=4$；
3. $2+1=3$；
4. $2+3=5$；
5. $3+1=4$；
6. $3+2=5$。

因此结果的期望为 $\frac{1}{6}(3+4+3+5+4+5)=4$。

【样例 2 输入】

```
3
_+_*_
```

【样例 2 输出】

```
666666677
```

【样例 2 解释】

将数字卡片 1,2,3 填入算式 _+_*_，可能得到 6 种不同的算式，结果分别如下：

1. $1+2\times 3=7$；
2. $1+3\times 2=7$；
3. $2+1\times 3=5$；
4. $2+3\times 1=5$；
5. $3+1\times 2=5$；
6. $3+2\times 1=5$。

因此结果的期望为 $\frac{1}{6}(7+7+5+5+5+5)=\frac{17}{3}$。因为 $3\times 666666677\equiv 17 \pmod{1000000007}$，所以 $\frac{17}{3}$ 对 10^9+7 取模的结果是 666666677。

【样例 3】

见选手目录下的 ***expr/expr3.in*** 与 ***expr/expr3.ans***。

该组样例的数据范围同第 4 个测试点。

【样例 4】

见选手目录下的 ***expr/expr4.in*** 与 ***expr/expr4.ans***。
该组样例的数据范围同第 7,8 个测试点。

【样例 5】

见选手目录下的 ***expr/expr5.in*** 与 ***expr/expr5.ans***。
该组样例的数据范围同第 14 个测试点。

【样例 6】

见选手目录下的 ***expr/expr6.in*** 与 ***expr/expr6.ans***。
该组样例的数据范围同第 16 个测试点。

【样例 7】

见选手目录下的 ***expr/expr7.in*** 与 ***expr/expr7.ans***。
该组样例的数据范围同第 21 个测试点。

【说明】

本题中，算术表达式的格式如下：（格式中 <、> 代表一个整体，并不是表达式的一部分）

< 表达式 > : < 运算数 1>< 运算符 1>< 运算数 2>< 运算符 2>...< 运算符 k-1>< 运算数 k>（k 为正整数）

其中，运算数或者是一个数（在本题中，算式中所有的数都被换成了 _），或者是 (< 表达式 >) 的形式，即包含在括号内的表达式；运算符为 +、-、* 之一，其中 +、-、* 分别代表算术加、减、乘运算。

计算时，按照先乘后加减的顺序计算，同级运算从左到右进行。对于表达式中的括号，应先计算括号内的结果；**括号可能有嵌套**，应从内到外计算括号内的表达式。

【提示】

1. 表达式中 - 不会作为负号（即 - 和 +、* 一样总是出现在两个运算数之间），也不会出现连续的两个 _（即不会有多个数连接形成一个数的情况）。
2. 请注意，n **不是**字符串 S 中 _ 的个数。

【子任务】

我们记 $|S|$ 为字符串 S 的长度，每个测试点的数据规模及特点如下表所示：

<table>
<tr><th>测试点编号</th><th>n</th><th>$|S|$</th><th>特殊性质 1</th><th>特殊性质 2</th></tr>
<tr><td>1</td><td rowspan="3">≤ 5</td><td>≤ 1</td><td rowspan="4">是</td><td rowspan="2">是</td></tr>
<tr><td>2</td><td rowspan="2">≤ 3</td></tr>
<tr><td>3</td><td>否</td></tr>
<tr><td>4</td><td rowspan="3">≤ 8</td><td rowspan="5">≤ 20</td><td rowspan="3">是</td></tr>
<tr><td>5</td><td rowspan="4">否</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td><td rowspan="2">≤ 10</td><td rowspan="19">否</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td><td rowspan="2">≤ 20</td><td rowspan="5">≤ 60</td><td>是</td></tr>
<tr><td>10</td><td>否</td></tr>
<tr><td>11</td><td>≤ 25</td><td>是</td></tr>
<tr><td>12</td><td rowspan="2">≤ 30</td><td rowspan="2">否</td></tr>
<tr><td>13</td></tr>
<tr><td>14</td><td>≤ 100</td><td rowspan="2">≤ 200</td><td rowspan="2">是</td></tr>
<tr><td>15</td><td>≤ 300</td></tr>
<tr><td>16</td><td>≤ 500</td><td>≤ 500</td><td>否</td></tr>
<tr><td>17</td><td>≤ 1000</td><td>≤ 2000</td><td rowspan="2">是</td></tr>
<tr><td>18</td><td>≤ 3000</td><td rowspan="2">≤ 5000</td></tr>
<tr><td>19</td><td rowspan="2">≤ 5000</td><td rowspan="7">否</td></tr>
<tr><td>20</td><td>$\le 10^4$</td></tr>
<tr><td>21</td><td>$\le 10^7$</td><td rowspan="3">≤ 500</td></tr>
<tr><td>22</td><td>$\le 10^8$</td></tr>
<tr><td>23</td><td>$\le 10^9$</td></tr>
<tr><td>24</td><td>$\le 10^7$</td><td rowspan="2">$\le 10^4$</td></tr>
<tr><td>25</td><td>$\le 10^9$</td></tr>
</table>

其中，**部分数据**具有的特殊性质意义如下：

- 特殊性质 1：字符串 S 中仅包含 `_`、`+`、`*` 三种字符；
- 特殊性质 2：保证答案是小于 10^9+7 的非负整数；对于这类数据，输出答案对 10^9+7 取模的结果和直接输出答案是一致的。