

NOIP2017 模拟赛 Day2

竞赛时间：2017 年 9 月 16 日上午 8：30 - 11：30

题目名称	括号匹配	计算数列	线路规划
可执行文件名	match	sequence	route
输入文件名	match.in	sequence.in	route.in
输出文件名	match.out	sequence.out	route.out
每个测试点时限	1s	1s	1s
内存限制	128MB	128MB	128MB
题目分值	100	100	100
测试点个数	10	10	10
单个测试点分值	10	10	10
题目类型	传统	传统	传统

说明：

- 1.代码长度限制为 100KB，编译时开启 -O2 开关。
- 2.请将自己的代码按照要求放入文件夹内。
- 3.请注意不要因为非技术因素导致程序无法正常通过数据，其中你要注意到的包括但不限于：
 - (1).内存使用情况。
 - (2).是否使用文件输入输出，文件输入输出的.in/.out 的文件名是否正确，源程序的文件名是否正确。源程序的文件名和.in/.out 的文件名是否有不可见字符，如果有，则认为文件名错误，不能得分。
 - (3).保存文件的路径是否正确。
 - (4).是否删除调试信息。
 - (5).是否能通过所有样例。
 - (6).输出格式是否正确。
 - (7).变量类型是否正确。

括号匹配

【问题描述】

定义一个仅包含括号“(”和“)”的括号序列为合法序列如下。

- 1、空串“”为合法序列。
- 2、如果串 A、B 均为合法序列，则 $C=A+B$ （即将 B 拼在 A 后面）也是一个合法序列。
- 3、如果串 A 为合法序列，则串 $C=“(”+A+“)”$ （即在 A 的外面套一层括号）也是一个合法序列。

例如，“()()()()”、“(())()”均是合法序列，而“()()”、“()()”则不是合法序列。

现在给你一个长度为 N 的仅包含括号“(”和“)”的括号序列 S，你每次操作可以交换其中两个括号的位置，求将 S 变成一个合法序列的最少操作次数，无解则输出 -1。

【输入】

第一行为一个正整数 N。

第二行为一个字符串 S。

【输出】

输出最少操作次数，无解则输出 -1。

【输入输出样例】

match.in	match.out
6)))(2

【样例解释】

)))(→ ())(→ (())

【数据范围】

对于 30% 的数据： $1 \leq N \leq 10$ 。

对于 60% 的数据： $1 \leq N \leq 1000$ 。

对于 100% 的数据： $1 \leq N \leq 100000$ 。

计算数列

【问题描述】

给定一个长度为 N 的数列 A 。

设 $Max(l, r) = \text{Max}\{A[l], A[l+1], \dots, A[r]\}$, $Gcd(l, r) = \text{gcd}\{A[l], A[l+1], \dots, A[r]\}$ 。

求 $\sum_{l=1}^n \sum_{r=l}^n Gcd(l, r) \times Max(l, r)$ 对 998244353 取模的结果。

【输入】

第一行为 1 个正整数 N 。

第二行为 N 个正整数 $A[i]$ 。

【输出】

输出答案对 998244353 取模的结果。

【输入输出样例】

sequence.in	sequence.out
6 2 4 6 3 8 4	303

【数据范围】

测试数据编号	数据范围	其他限制
1 - 3	$1 \leq N \leq 1000$	无
4 - 5	$1 \leq N \leq 152501$	保证数列 A 由 N 个互不相同的质数组成
6 - 7		保证数列 A 为随机生成
8 - 10		无

对于 100% 的数据： $1 \leq A[i] \leq 10^9$ 。

线路规划

【问题描述】

小 S 是 Q 国交通部的部长，他最近在为一批物资的运输线路发愁。

Q 国有 N 个城市，由 $N - 1$ 条双向道路连接，其中第 i 条道路连接了城市 i 和城市 $i + 1$ ，这些道路通过所花费的时间均为 1。

现在有 M 条运输物资的线路，第 i 条线路是从 $u[i]$ 号城市向 $v[i]$ 号城市运输的。

现在因为 Q 国执行了圣光的教义，诸神可以在两个城市间建立一座传送阵。这座传送阵可以连接任意两座城市，而且从其中一座城市可以瞬间移动到另一座城市。

现在小 S 想要尽快将这些运输方案完成，他想知道最优情况下最后到达目的地的运输方案最早是多少。

【输入】

第一行为 2 个正整数 N 、 M 。

接下来 m 行，每行两个正整数 $u[i]$ 、 $v[i]$ 。

【输出】

输出最优情况下最后到达目的地的运输方案的时间。

【输入输出样例 1】

route1.in	route1.out
5 2	1
1 3	
2 4	

【样例 1 解释】

在 2 号城市与 3 号城市建立传送阵，这样运输方案时间最长的只需要 1 点时间就可以了。

【输入输出样例 2】

route2.in	route2.out
9 8	3
3 5	
4 1	
8 4	
8 4	
8 5	
8 9	
4 9	
8 2	

【数据范围】

测试数据编号	N、M
1	= 100
2	= 200
3	= 501
4	= 2501
5	= 5210
6	= 5210
7	= 52501
8	= 152501
9	= 252501
10	= 525010

对于 100%的数据： $1 \leq N、M \leq 525010$ ， $1 \leq u[i]、v[i] \leq N$ 。