

رمزهای جان سخت

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

همواره برای انتخاب یک رمز برای استفاده‌های مختلف انسان‌ها در حال چالش با خود هستند!! یک نظریه وجود دارد که می‌گوید در میان رمزهایی که کاملاً متشکل از ارقام هستند (یعنی شامل کاراکتر نیستند) آن دسته از رمزهایی که جان سخت (!) باشند دارای امنیت بالایی هستند.

یک رمز N رقمی جان سخت است اگر عددی اول باشد و اگر از سمت راست آن شروع به حذف ارقام آن بکنیم همواره عدد باقی‌مانده نیز عددی اول باشد. مثلاً رمز 2399 یک رمز جان سخت است زیرا اعداد ۲۳۹۹ و ۲۳۹ و ۲۳ و ۲ همگی اول هستند. همچنین 3137 نیز یک رمز جان سخت است.

برنامه ای بنویسید که عدد N را بگیرد و تمام رمزهای جان سخت به طول N را چاپ کند.

دقت کنید که عدد ۱ عدد اول نیست.

ورودی

یک خط شامل عدد N

$$1 \leq N \leq 8$$

خروجی

رمزهای جان سخت به طول N که به صورت صعودی مرتب شده‌اند و همچنین در هر خط یک عدد وجود دارد.

مثال

ورودی نمونه ۱

3

خروجی نمونه ۱

- 233
- 239
- 293
- 311
- 313
- 317
- 373
- 379
- 593
- 599
- 719
- 733
- 739
- 797

وارونگی

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

باقی‌مانده تعداد وارونگی‌ها در یک آرایه A به طول N را بر عدد 100 000 چاپ کنید. یک وارونگی در یک آرایه به صورت زیر تعریف می‌شود:

$$1 \leq i, j \leq N, i > j, A_i < A_j$$

ورودی

در خط اول N که تعداد اعضای آرایه است به شما داده می‌شود. در N خط بعدی در هر خط یک عدد که عنصر i ام آرایه است به شما داده می‌شود.

$$1 \leq N \leq 200\,000$$

خروجی

یک خط که شامل تعداد وارونگی‌های آرایه است که بر عدد 100 000 باقی‌مانده گرفته شده‌اند.

مثال

ورودی نمونه

3
2
3
1

خروجی نمونه

2

ضیافت دانشگاه امیرکبیر

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

برای ضیافت دانشگاه امیرکبیر به مناسبت آغاز سریع ترم جدید تعداد N لامپ کم‌مصرف دوستدار طبیعت تهیه کرده‌ایم که با شماره‌های یک تا N شماره‌گذاری شده‌اند. لامپ‌ها را به کلیدهای زیر متصل کرده‌ایم:

- کلید ۱: وقتی این کلید زده شود تمامی لامپ‌ها تغییر وضعیت می‌دهند و اگر خاموش هستند روشن می‌شوند و اگر روشن باشند خاموش می‌شوند.
- کلید ۲: تمامی لامپ‌ها با شماره فرد تغییر وضعیت می‌دهند.
- کلید ۳: تمامی لامپ‌ها با شماره زوج تغییر وضعیت می‌دهند.
- کلید ۴: تمامی لامپ‌ها با شماره $3k + 1$ تغییر وضعیت می‌دهند مانند ۱، ۴، ۷، ...

از طرفی برای کنترل استهلاک ضیافت، یک شمارنده در دفتر دانشگاه وجود دارد که تعداد کل فشردن کلیدها را کنترل می‌کند. هنگامی که ضیافت آغاز می‌شود کل لامپ‌ها روشن هستند و عدد شمارنده برابر با صفر است.

در طول ضیافت کلیدها به ترتیبی فشرده شده‌اند و عدد شمارنده به C افزایش یافته است. وضعیت نهایی تعدادی از لامپ‌ها را می‌دانیم. می‌خواهیم بدانیم سایر لامپ‌ها چه وضعیت‌هایی ممکن است داشته باشند.

اعداد N و C و وضعیت نهایی برخی از لامپ‌ها داده می‌شود. برنامه‌ای بنویسید که تمامی حالات نهایی ممکن با C بار فشردن کلیدها و وضعیت نهایی داده شده را مشخص کند. حالات نهایی حساب شده نباید تکراری باشند.

ورودی

- خط اول: عدد N که تعداد لامپ‌هاست.
- خط دوم: عدد C که تعدادن فشردن کلیه کلیدهاست. ($1 \leq C \leq 10000$)

- خط سوم: لامپ‌هایی که باید در آخر روشن باشند که در آخر خط به منظور نشان دادن پایان ورودی 1- می‌آید.
- خط چهارم: لامپ‌هایی که باید در آخر خاموش باشند که در آخر خط به منظور نشان دادن پایان ورودی 1- می‌آید.

خروجی

شامل تمامی حالات نهایی ممکن با ویژگی‌های ورودی است. هر خط شامل N کاراکتر است که وضعیت لامپ‌ها را در آن حالت بیان می‌کند. اگر لامپی خاموش باشد عدد 0 و اگر روشن باشد عدد 1 را می‌گذاریم. اگر هیچ حالتی وجود نداشت باید در خروجی عبارت IMPOSSIBLE چاپ شود.

مثال

ورودی نمونه

10
1
-1
7 -1

خروجی نمونه

0000000000
0101010101
0110110110