

باسمه تعالی

شانزدهمین دوره‌ی آموزشی المپیاد کامپیوتر

امتحان فاینال عملی دوم

چهارشنبه ۱۵ شهریورماه ۱۳۸۵

اويس قرن، نصیری شرق

وقت: پنج ساعت

توصیه‌های ایمنی

- در خط اول هر یک از برنامه‌های تان نام خود را با فرمت روبه‌رو قرار دهید: `//name: YourNameHere`
مثال: `//name: Freddie Mercury`
فراموش کردن این مهم در هر یک از برنامه‌های شما، موجب از دست دادن تمام نمره‌ی آن سؤال خواهد شد. دقت کنید که اولین فاصله‌ی خالی این سطر، بعد از : می‌باشد.
- توصیه می‌شود قبل از ترک جلسه‌ی امتحان حتماً یک‌بار تست‌های برگه^۱ را به برنامه‌های تان بدهید. نیز حتماً یک‌بار دیگر برنامه‌های تان را هم‌گردانی^۲ کنید.
- وقت امتحان پنج ساعت است و پس از اتمام وقت، برق سالن قطع خواهد شد. حداقل زمان حضور در جلسه نیز یک ساعت می‌باشد.
- در پانزده دقیقه‌ی ابتدایی به هیچ سؤالی پاسخ داده نمی‌شود. در این زمان حتماً صورت هر دو سؤال را بخوانید. سؤالات شما، صرفاً در باب رفع ابهام، فقط در یک ساعت اول جواب داده خواهند شد.
- نام کاربری شما در این امتحان **Final2** و رمز عبور آن **inammesleghabli** می‌باشد. در طول امتحان فقط از همین کاربر استفاده کنید.
- برنامه‌های تان را در شاخه‌ی ~ با نام‌های **pebble.cpp** و **snail.cpp** ذخیره کنید. دقت کنید که هنگام جمع‌آوری فایل‌ها تنها وجود دو فایل `/home/Final2/pebble.cpp` و `/home/Final2/snail.cpp` در رایانه‌ی شما بررسی خواهد شد و بعداً هیچ عذری در این زمینه پذیرفتنی نیست.
- محدودیت زمانی برنامه‌های شما یک ثانیه (روی سرور) برای سؤال اول و شش ثانیه برای سؤال دوم در نظر گرفته شده است. بدیهی است پس از این مدت برنامه‌ی شما کلاً پا خواهد شد. هیچ‌گونه محدودیت حافظه برای برنامه‌های شما اعمال نخواهد شد؛ با این حال توجه داشته باشید که استفاده از حافظه‌ی زیاد باعث کند شدن برنامه می‌شود.
- یک‌بار دیگر یادآوری می‌شود که از آنجایی که این آزمون، یک امتحان ۶ درصدی فاینال شماست و هر یک از سؤالات آن معادل ۳ درصد نمره‌ی اصلی می‌باشند، فراموش کردن بی‌دقتی در ذخیره‌سازی، خروجی نوشتن و سایر موارد به‌راحتی باعث از دست دادن این نمره می‌شوند. فلذا دقت لازم را به‌عمل آورید.
- این آخرین آزمون شما در دوره‌ی تابستانه‌ی سال ۱۳۸۵ است؛ یادش به‌خیر!

۱ Sample input
۲ compile

۱ مهره‌های پرتحرک ۱۰۰ نمره

گراف جهت‌دار بی‌دور G به شما داده شده است. در ثانیه‌ی صفر روی هر کدام از رئوس این گراف تعدادی مهره قرار گرفته است. در هر لحظه برای هر کدام از رئوس گراف عمل زیر صورت می‌گیرد:

در صورتی که تعداد مهره‌های روی آن رأس، کم‌تر از درجه‌ی خروجی‌اش نباشد، آن رأس به هر کدام از رأس‌هایی که به آن‌ها یال دارد، یکی از مهره‌هایش را می‌دهد.

دقت کنید که این عمل در هر ثانیه به‌طور هم‌زمان روی همه‌ی رئوس گراف G انجام می‌شود. به عبارت دیگر ممکن است در یک ثانیه رأس v به رأس‌هایی که به آن‌ها یال می‌دهد مهره بدهد و از رأس‌هایی که به او یال دارند مهره دریافت کند. بدیهی است که اگر درجه‌ی خروجی راس v صفر باشد، هیچ گاه مهره‌ای از آن خارج نمی‌شود. می‌توان ثابت کرد که با انجام این عمل به تعداد زیاد سرانجام لحظه‌ای فرا می‌رسد که هیچ مهره‌ای حرکت نمی‌کند؛ این زمان t را لحظه‌ی نهایی می‌نامیم.

مسئله

برنامه‌ای بنویسید که گراف G و تعداد مهره‌هایی که در ثانیه‌ی صفر در هر رأس آن وجود دارد را بخواند و تعداد مهره‌های موجود در هر رأس را در لحظه‌ی نهایی بنویسد.

ورودی: فایل pebble.in

در سطر نخست ورودی ابتدا عدد n ، تعداد رئوس گراف G و سپس عدد e ، تعداد یال‌های گراف آمده است. در n سطر بعدی در هر سطر یک عدد آمده است به‌طوری که عدد موجود در سطر $i + 1$ ، i ام، تعداد مهره‌هایی است که در حالت اولیه در رأس i ام گراف G قرار دارند. در e سطر بعدی در هر سطر به‌ترتیب دو عدد x و y آمده است که نشان‌دهنده‌ی یک یال جهت‌دار از رأس x به رأس y است.

خروجی: فایل pebble.out

در تنها سطر فایل خروجی شما بایستی n عدد بنویسید به‌طوری که عدد i ام موجود در این سطر نشان‌دهنده‌ی تعداد مهره‌هایی است که در لحظه‌ی نهایی در رأس i ام گراف قرار خواهند داشت. اعداد را با فاصله‌ی خالی از هم جدا کنید.

محدودیت‌ها

- $1 \leq n \leq 50,000$.
- $1 \leq e \leq 200,000$.
- می‌توانید فرض کنید که به ازای هر زوج مرتب (x, y) حداکثر یک یال جهت‌دار از x به y وجود دارد.
- مجموع تعداد مهره‌هایی که روی رئوس گراف قرار دارد کمتر از 2^{31} است.

۳ DAG

۴ که زودترین ثانیه‌ای است که در آن هیچ مهره‌ای جابه‌جا نمی‌شود

مثال

pebble.in	pebble.out
3 2 2 0 0 1 2 2 3	0 0 2

۲ حلزون ۱۰۰ نمره

یک کرم حلزونی شکل بسیار دراز قصد دارد روی یه ماتریس از اعداد، حمام آفتاب بگیرد! می‌دانیم کرم برای دراز کشیدن روی ماتریس، ابتدا از مسافتی دور به سمت یکی از خانه‌های مجاور چهار ضلع ماتریس حرکت می‌کند تا بتواند در هر حرکت بعد وارد ماتریس شود؛ پس از آن در هر مرحله، در صورت امکان یکی از اعمال زیر را انجام می‌دهد:

- یا یک واحد در امتداد جهت حرکت قبلی خود به جلو می‌رود؛ به‌عنوان مثال، اگر در حرکت قبل به سمت راست رفته، در این حرکت هم به سمت راست می‌رود.
- یا در امتداد جهت حرکت قبل خود، ابتدا ۹۰ درجه به راست می‌چرخد و سپس یک واحد جلو می‌رود؛ به‌عنوان مثال، اگر در حرکت قبل به سمت راست رفته، در این حرکت به سمت پایین می‌رود.
- یا متوقف می‌شود.

می‌دانیم کرم هیچ‌گاه از روی خودش عبور نخواهد کرد و از ماتریس نیز خارج نخواهد شد. بدیهی است که در پایان کار، مسیر حرکت کرم یک مارپیچ حلقوی در جهت حرکت عقربه‌های خواهد بود. اکنون با فرض این‌که طول کرم بسیار زیاد است و هیچ‌وقت دم‌ش وارد ماتریس نمی‌شود، به کرم کمک کنید تا طوری حرکت کند که مجموع اعدادی که روی آن‌ها قرار گرفته‌است، بیشینه باشد.

ورودی: فایل snail.in

در سطر اول فایل ورودی، دو عدد n و m که به ترتیب تعداد سطرها و ستون‌های ماتریس است نوشته شده است. سپس در n سطر بعدی ورودی، اعداد سطرهای ماتریس به ترتیب آمده است.

خروجی: فایل snail.out

در تنها سطر خروجی، بیش‌ترین مجموع خانه‌های طی شده توسط کرم را بنویسید.

محدودیت‌ها

- $1 \leq n, m \leq 50$
- $-2 \times 10^5 \leq \text{اعداد ماتریس} \leq 2 \times 10^5$

مثال

snail.in	snail.out
<pre> 4 4 1 0 0 0 1 1 1 1 1 -1 -1 1 1 1 1 1 </pre>	<pre> 11 </pre>