

باغ استوایی (Tropical Garden)

Somhed که یک گیاه‌شناس است معمولاً گروه‌های دانش‌آموزی را به یکی از بزرگ‌ترین باغ‌های استوایی تایلند می‌برد. فضای این باغ شامل N تقاطع (با شماره‌های ۰ تا $N - 1$) و M راه است. هر راه دو تقاطع متفاوت را مستقیماً به هم وصل می‌کند و می‌توان از هر دو طرف آن را پیمود. هر تقاطع حداقل یک راه متصل به خود دارد. این راه‌ها شامل مجموعه‌های گیاهی زیبایی هستند که Somhed علاقه‌مند به دیدن آن‌هاست. هر گروه مسیر خود را از هر کدام از تقاطع‌ها می‌تواند شروع کنند.

Somhed گیاهان زیبایی گرمسیری را دوست دارد. لذا، از هر تقاطعی او و دانش‌آموزانش راهی را انتخاب می‌کنند که زیباتر از همه است، مگر این‌که هم این زیباترین راه، همین آخرین راهی باشد که آمده‌اند و هم حداقل یک راه دیگر جز این زیباترین راه وجود داشته باشد؛ که در آن صورت، آنها دومین زیباترین راه را انتخاب می‌کنند. واضح است که در صورت نداشتن چاره (هیچ راه دیگری نباشد) آن‌ها از همان راهی که آمده‌اند برمی‌گردند. توجه کنید که چون Somhed یک گیاه‌شناس حرفه‌ایست هیچ دو راهی از نظر او زیبایی یکسانی ندارند.

دانش‌آموزان Somhed علاقه‌ی زیادی به گیاهان ندارند؛ ولی خیلی مایل هستند که ناهار را در رستوران خوبی که در تقاطع شماره P قرار دارد، بخورند. Somhed می‌داند که دانش‌آموزانش بعد از پیمودن دقیقاً K راه، گرسنه می‌شوند که K ممکن است برای هر گروه، عددی متفاوت باشد. Somhed می‌خواهد بداند چند راه متفاوت می‌تواند برای هر گروه انتخاب کند با این شرط که:

- هر گروه می‌تواند از هر تقاطعی کارش را شروع کند
- راه بعدی باید بر اساس قوانین بالا انتخاب شود
- هر گروه باید بعد از پیمودن دقیقاً K راه در تقاطع شماره P باشد.

توجه کنید که هر گروه می‌تواند چندین بار از تقاطع شماره P عبور کند؛ تنها نکته مهم این است که باید در انتها حتماً در تقاطع شماره P باشند.

وظیفه شما

با داشتن اطلاعات مربوط به راه‌ها و تقاطع‌ها شما باید جواب را برای هر کدام از Q گروه پیدا کنید. یعنی برای هر کدام از Q مقدار K ی مربوطه را باید بیابید.

روال $\text{count_routes}(N, M, P, R, Q, G)$ را بنویسید که پارامترهای زیر را می‌پذیرد

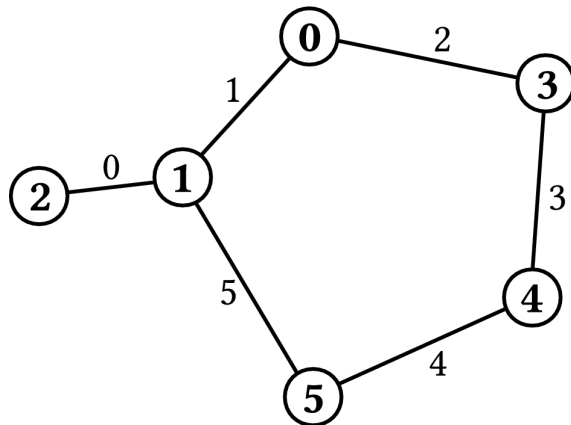
- N : تعداد تقاطع‌ها. تقاطع‌ها با شماره‌های ۰ تا $N - 1$ مشخص می‌شوند
- M : تعداد راه‌ها. راه‌ها با شماره‌های ۰ تا $M - 1$ مشخص می‌شوند. راه‌ها به ترتیب نزولی زیبایی داده شده‌اند، یعنی راه i از راه $i + 1$ زیباتر است (برای هر مقدار $0 \leq i < M - 1$)
- P : تقاطعی که رستوران محبوب در آن قرار دارد.

- R : آرایه‌ای دو بعدی که راه‌ها را نشان می‌دهد. برای هر $0 \leq i < M$ ، راهی i تقاطع‌های $R[i][0]$ و $R[i][1]$ را به هم وصل می‌کند. به یاد داشته باشید که هر راه، دو تقاطع متفاوت را به هم وصل می‌کند و هیچ دو راهی دو تقاطع یکسان را به هم وصل نمی‌کنند.
 - Q : تعداد گروه‌های دانش‌آموزان.
 - G : یک آرایه‌ی یک بُعدی شامل اعداد K . برای $0 \leq i < Q$ ، $G[i]$ تعداد راه‌هایی است که گروه i ام می‌پیماید.
- برای $0 \leq i < Q$ ، برنامه‌ی شما باید تعداد مسیرهای ممکن با دقتاً $G[i]$ راه را پیدا کند که با پیمودن آن‌ها، گروه i ام به تقاطع P می‌رسد. برای هر i ، برنامه شما باید روال $\text{answer}(X)$ را برای گزارش این‌که تعداد مسیرها X است صدا بزند.
- ترتیب جوابها باید مطابق با ترتیب سؤالات باشد. اگر هیچ مسیری وجود نداشت، برنامه شما باید $\text{answer}(0)$ را فراخوانی کند.

مثال‌ها

مثال اول

شکل روبرو را ببینید که در آن $N = 6, M = 6, P = 0, Q = 1$ و $G[0] = 3$ است.



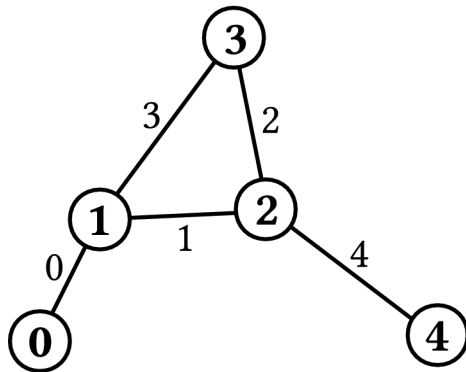
$$R = \begin{matrix} 1 & 2 \\ 0 & 1 \\ 0 & 3 \\ 3 & 4 \\ 4 & 5 \\ 1 & 5 \end{matrix}$$

توجه کنید که راه‌ها بر اساس ترتیب نزولی زیبایی آمده‌اند به این معنی که راه ۰ زیباترین، راه ۱ دومین زیباترین و ... است. در این مثال فقط دو مسیر به طول ۳ وجود دارد: $0 \rightarrow 1 \rightarrow 2 \rightarrow 1$ و $0 \rightarrow 3 \rightarrow 4 \rightarrow 5$. اولین مسیر از تقاطع ۱ شروع می‌شود. زیباترین راه متصل به آن به تقاطع ۲ ختم می‌شود. گروه، در اینجا چاره‌ای غیر از بازگشت به تقاطع ۱ ندارند. حالا این گروه به جای راه ۰ راه ۱ را انتخاب کرده که آنها را به تقاطع $P = 0$ می‌برد.

بنابراین برنامه روال $\text{answer}(2)$ را فراخوانی می‌کند.

مثال دوم

شکل زیر را ببینید. در این مثال آن $N = 5, M = 5, P = 2, Q = 2$ و $G[0] = 3$ و $G[1] = 1$ است.



$$R = \begin{matrix} 1 & 0 \\ 1 & 2 \\ 3 & 2 \\ 1 & 3 \\ 4 & 2 \end{matrix}$$

برای گروه اول، فقط یک مسیر هست که طی ۳ مرحله به تقاطع ۲ می‌رسد:

$$1 \rightarrow 0 \rightarrow 1 \rightarrow 2$$

برای گروه دوم، دو مسیر هست که طی یک مرحله به تقاطع ۲ می‌رسند:

$$3 \rightarrow 2 \text{ و } 4 \rightarrow 2$$

لذا در پیاده‌سازی درست `count_routes` باید ابتدا `answer(1)` را فراخوانی کند (برای جواب به گروه اول) و سپس `answer(2)` را برای جواب به گروه دوم فراخوانی کند.

زیرمسئله‌ها

زیرمسئله شماره سه (۳۱ امتیاز)

$$2 \leq N \leq 150,000 \quad \bullet$$

$$1 \leq M \leq 150,000 \quad \bullet$$

$$1 \leq Q \leq 2,000 \quad \bullet$$

- هر عنصر آرایه‌ی G یک عدد صحیح بین ۱ تا $1,000,000,000$ (شامل خود این دو عدد) است.

زیرمسئله شماره یک (۴۹ امتیاز)

$$2 \leq N \leq 1,000 \quad \bullet$$

$$1 \leq M \leq 10,000 \quad \bullet$$

$$Q = 1 \quad \bullet$$

- هر عنصر آرایه‌ی G یک عدد صحیح بین ۱ تا ۱۰۰ (شامل خود این دو عدد) است.

زیرمسئله شماره دو (۲۰ امتیاز)

$$2 \leq N \leq 150,000 \quad \bullet$$

$$1 \leq M \leq 150,000 \quad \bullet$$

$$Q = 1 \quad \bullet$$

- هر عنصر آرایه‌ی G یک عدد صحیح بین ۱ تا $1,000,000,000$ (شامل خود این دو عدد) است.

جزئیات پیاده‌سازی

محدودیت‌ها

- محدودیت زمانی CPU: ۵ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت
- توجه: هیچ محدودیت صریحی روی اندازه‌ی حافظه‌ی پشته (stack) استفاده شده وجود ندارد. میزان حافظه‌ی استفاده شده در پشته به‌عنوان بخشی از حافظه‌ی مصرفی محاسبه می‌شود.

واسط API

- فولدر پیاده‌سازی: garden/
- فایل‌هایی که توسط شرکت‌کننده می‌بایست پیاده‌سازی شوند: garden.c یا garden.cpp یا garden.pas
- واسط‌های شرکت‌کننده: garden.h یا garden.pas
- واسط مصحح: gardenlib.h یا gardenlib.pas
- مصحح نمونه: grader.c یا grader.cpp یا grader.pas
- ورودی مصحح نمونه: grader.in.1، grader.in.2 و ...
- توجه: مصحح نمونه، ورودی را به فرمت زیر می‌خواند:
 - خط اول: مقادیر M ، N و در نهایت P .
 - خط ۲ الی $M + 1$ ام: توضیحات راه‌ها؛ یعنی خط $i + 2$ ام شامل $R[i][0]$ و $R[i][1]$ (جداشده با یک فاصله)، برای $0 \leq i < M$ است.
 - خط $M + 2$ ام: مقدار Q .
 - خط $M + 3$ ام: آرایه‌ی G در قالب یک رشته از اعداد صحیح که با فاصله جدا شده‌اند.
 - خط $M + 4$ ام: آرایه‌ی جواب‌های مورد انتظار، در قالب یک رشته از اعداد صحیح که با فاصله جدا شده‌اند.
- خروجی مورد انتظار برای ورودی مصحح نمونه: grader.expect.1، grader.expect.2 و ...
برای هر مساله، هر کدام از این فایلها باید دقیقاً شامل عبارت «**Correct.**» باشد.