

به نام او



دوره ی پایانی المپیاد کامپیوتر ایران
فرودین ۱۳۸۷
آزمون پنجم

Click

کلیک

گراف G ، n راس و m یال دارد. این گراف طوقه ندارد (یعنی رئوس دو سر هیچ یالی یکی نیست) و بین هر دو راسی حداکثر یک یال وجود دارد. به عبارت دیگر G ساده است. رئوس G با اعداد 0 تا $n-1$ (شامل این دو عدد) شماره گذاری شده اند. روی هر یال G یک عدد صحیح بین 0 تا $k-1$ نوشته شده است. شما به قسمتی از اطلاعات این گراف از طریق یک کتابخانه دسترسی دارید و می توانید اعداد یال های این گراف را به نحوی خاص که توضیح داده خواهد شد عوض کنید. در این کتابخانه سه تابع به نام های $libN$ ، $libK$ و $click$ جهت کسب اطلاعات از گراف و ایجاد تغییر در آن در نظر گرفته شده اند. تابع $libN()$ و $libK()$ به ترتیب n و k را برمی گردانند. تابع $click(i)$ روی راس i کلیک می کند. با این عمل یک واحد به اعداد روی یال های متصل به راس i اضافه می شود به جز یال های با شماره ی $k-1$ که تبدیل به صفر می شوند (از 0 به 1 ، از 1 به 2 ، ...، از $k-1$ به 0). تعداد یال هایی که پس از انجام این عمل صفر می شوند به عنوان خروجی تابع برگردانده می شود. زمان اجرای تابع $click(i)$ متناسب با تعداد یال های متصل به راس i است. به عبارت دیگر این تابع در زمان $\Theta(d_i)$ اجرا می شود که d_i درجه ی راس i است.

گراف های ورودی در کتابخانه نهفته شده اند و برنامه ی شما از ورودی استاندارد چیزی نمی خواند. یازده عدد گراف ورودی در کتابخانه با شماره های 0 تا 10 طراحی شده اند. برای بارگذاری^۱ تست t ام باید از تابع $reset(t)$ استفاده کنید.

هدف در این مسئله این است که شما با استفاده از کتابخانه ای که در اختیارتان گذاشته شده یال های هر یک از یازده گراف و اعدادی که در ابتدا روی این یال ها قرار داشته را بدست آورید. توجه کنید که تابع $click$ اعداد روی بعضی یال ها را از حالت اولیه تغییر می دهد.

از شما انتظار می رود در نهایت ادب و نزاکت، خروجی خواسته شده را در یازده فایل خروجی با نام های $click0.out$ ، ...، $click10.out$ قرار داده و همه ی این فایل ها را در یک پوشه با نام $click$ قرار داده، سپس پوشه را فشرده کرده و فایل $click.zip$ یا $click.tgz$ را ارسال کنید. در صورتی که فقط فایل بعضی از تست ها در فرستاده شده باشد، داوری فقط برای همان تست ها انجام می شود.

برای دیدن جزئیات بیشتری از توابع کتابخانه به قسمت «کتابخانه» در صورت مسئله توجه کنید.

مسئله

برنامه ای بنویسید که

^۱Load

- تست مورد نظر را بارگذاری کند و n و k را از کتابخانه پیوندی بگیرد؛
- با استفاده از کتابخانه پیوندی یال‌های گراف و اعدادی که در ابتدا روی آن‌ها قرار دارند را پیدا کند؛
- مشخصات گراف اولیه را در فایل مربوطه ذخیره کند.

خروجی

در خط اول، m (تعداد یال‌های گراف) را چاپ کنید. در m خط بعدی باید یال‌های گراف را بدین صورت چاپ کنید: ابتدا دو سر آن یال و سپس عدد روی آن در گراف اولیه را بنویسید. ترتیب یال‌ها و ترتیب دو سر هر یال در خروجی اهمیت ندارد، اما یال تکراری در خروجی نباید چاپ کنید.

شیوه‌ی ارزیابی

اگر یال‌های گراف خروجی شما و اعداد روی آن درست باشد نمره‌ی کامل آن تست به شما داده می‌شود. در صورتی که گراف خروجی شما یال‌هایش همان یال‌های خروجی صحیح باشد، اما عدد روی یک یا چند یال را اشتباه بنویسید، ۴۰ درصد نمره آن تست به شما تعلق می‌گیرد.

کتابخانه

`void reset(int t)` با فراخوانی این تابع گراف تست t ام بارگذاری می‌شود. قبل از اولین بار صدا کردن این تابع، به طور پیش فرض تست صفرم بارگذاری شده است. اگر $0 \leq t \leq 10$ نباشد، این تابع کاری روی گراف انجام نمی‌دهد.

اخطار: این تابع برای بعضی تست‌ها (به خصوص تست‌های آخر) نسبتاً کند است. به عنوان مثال `reset(10)` روی یک کامپیوتر معمولی چند میلی ثانیه زمان می‌برد.

`int libN()` این تابع n ، تعداد رئوس گراف را برمی‌گرداند.

`int libK()` این تابع k را برمی‌گرداند.

`int click(int i)` اگر $0 \leq i < n$ ، روی راس i ام کلیک می‌کند و تعداد یال‌های مجاور i که پس از کلیک کردن مقدارشان صفر شده است را برمی‌گرداند. در غیر این صورت کاری روی گراف انجام نمی‌دهد و ۱- برمی‌گرداند.

فایل‌های `libclick.h`، `libclick.o` و `sclick.{c,cpp}` را می‌توانید از بخش Download در سیستم مسابقه برداشت کنید.

تعریف توابعی که کتابخانه در اختیارتان می‌گذارد، در فایل `libclick.h` قرار دارد.

برای کامپایل کردن برنامه‌تان (در این جا با نام `click.cpp`) با `g++` فایل `libclick.o` را پیش برنامه‌تان کپی کرده و از دستور

```
g++ -O2 -static -lm -o click click.cpp libclick.o
```

استفاده کنید. در فایل `sclick.{c,cpp}` نمونه‌ای از کار کردن با این کتابخانه پیاده‌سازی شده است. توضیحات مربوط به این برنامه در بخش «تعامل نمونه» آمده است.

محدودیت‌ها

- $4 \leq n \leq 1000$
- $2 \leq k \leq 4000$
- $1 \leq m \leq 2000$
- حجم خروجی‌ای که به سرور ارسال می‌کنید باید کمتر از ۱۰۰ کیلوبایت باشد، در غیر این صورت سرور آن را قبول نمی‌کند.
- از آنجا که در این مسئله باید فقط خروجی‌ها را برای ما ارسال کنید، لزومی ندارد که اجرای برنامه‌ی شما برای همه‌ی تست‌ها مدت زمان کمی طول بکشد. حل همه‌ی تست‌ها ممکن است چندین دقیقه طول بکشد.

تعامل نمونه

توضیحات	فراخوانی
<code>reset(0);</code>	گراف تست صفرم را بارگذاری می‌کند.
<code>int n=libN();</code>	مقدار <code>n</code> برابر خروجی تابع یعنی 4 می‌شود.
<code>int k=libK();</code>	مقدار <code>k</code> برابر خروجی تابع یعنی 3 می‌شود.
<code>int a=click(0);</code>	روی راس صفرم کلیک می‌کنیم. مقدار <code>a</code> برابر خروجی تابع یعنی 1 می‌شود.
<code>int b=click(3);</code>	روی راس سوم کلیک می‌کنیم. مقدار <code>b</code> برابر خروجی تابع یعنی 0 می‌شود.
<code>int c=click(1);</code>	روی راس یکم کلیک می‌کنیم. مقدار <code>c</code> برابر خروجی تابع یعنی 2 می‌شود.
<code>int d=click(2);</code>	روی راس دوم کلیک می‌کنیم. مقدار <code>d</code> برابر خروجی تابع یعنی 0 می‌شود.
<code>int e=click(3);</code>	روی راس سوم کلیک می‌کنیم. مقدار <code>e</code> برابر خروجی تابع یعنی 1 می‌شود.
<code>int f=click(1);</code>	روی راس یکم کلیک می‌کنیم. مقدار <code>f</code> برابر خروجی تابع یعنی 0 می‌شود.

خروجی نمونه

خروجی صحیح مربوط به تست شماره‌ی صفر (`click0.out`) در جدول زیر آمده است. (البته هر ترتیب دیگری از یال‌ها نیز صحیح است.)

4
1 0 1
2 1 2
3 0 2
3 2 0