



Game

آزمون شبیه ساز فاینال اول وال شهریور ۱۴۰۵ آزمون عملی دوم

بازی (Game)

محدودیت زمانی: 1 ثانیه

محدودیت حافظه: 256 مگابایت

محمدمسام و امیرحسین پس از فراز و نشیب های بسیار تصمیم گرفتند روی یک گراف با یکدیگر بازی انجام دهند تا خستگی هایشان از تن به در شود. در ابتدا محمدمسام یک گراف n راسی بدون یال دارد. بازی $n - 1$ مرحله دارد و به این شکل است که در هر مرحله ابتدا محمدمسام یک یال بین دو راس رسم می کند به طوری که آن دو راس در دو مولفه متفاوت باشند، سپس امیرحسین با پرسیدن چند سوال از محمدمسام باید یال جدید رسم شده توسط محمدمسام را پیدا کند (یال های قبلی را از مراحل قبل می داند). امیرحسین در هر پرسش می تواند دو مجموعه راس مجزا از رئوس به محمدمسام بدهد و محمدمسام در ازای این دو مجموعه به امیرحسین می گوید که آیا یالی وجود دارد که یک سر آن در مجموعه اول و یک سر آن در مجموعه دوم باشد یا خیر (لزومی ندارد اجتماع این دو مجموعه تمامی رئوس را شامل شود اما اشتراک این دو مجموعه حتما باید تهی باشد). از آنجا که امیرحسین تا حد مرگ خسته است شما باید به جای او با محمدمسام بازی کنید و با پرسیدن حداکثر m سوال از محمدمسام یال های رسم شده توسط او را پیدا کنید.

پیاده سازی

در این سوال شما اجازه خواندن از ورودی یا نوشتن در خروجی را ندارید. در صورت هرگونه تلاش از خواندن از ورودی یا نوشتن در خروجی نمره شما ۰ خواهد شد. تنها راه شما برای تبادل اطلاعات، توابعی است که در ادامه معرفی میشوند. در ابتدای کد خود از "game.h" استفاده کنید. همچنین تاکید می شود که کد شما نباید دارای تابع main باشد.

■ `int query(int sa, int sb, int a[], int b[])`

این تابع از قبل پیاده سازی شده است و شما می توانید آن را برای پرسیدن سوال از محمدمسام فرا بخوانید و در ازای فراخواندن این تابع محمدمسام به شما بگوید آیا یالی بین دو مجموعه داده شده وجود دارد یا خیر (اگر وجود داشته باشد تابع ۱ و اگر نه ۰ خروجی می دهد). sa اندازه مجموعه اول، آرایه a رئوس مجموعه اول، sb اندازه مجموعه دوم و آرایه b رئوس مجموعه دوم خواهد بود، توجه کنید که دو آرایه a و b نباید عضو مشترک داشته باشند.

■ `void setRoad(int a, int b)`

این تابع از قبل پیاده سازی شده است، هنگامی که با سوال هایتان یال رسم شده توسط محمدمسام را پیدا کردید با فراخوانی این تابع یال را به او بگویید، a و b دو سر یال رسم شده هستند. شما در ازای فراخواندن این تابع جوابی از محمدمسام دریافت نمی کنید.

■ `void run(int n)`

شما باید این تابع را پیاده سازی کنید. در هر تست این تابع یک بار فراخوانی میشود و عدد n که تعداد رئوس گراف است به شما داده میشود. شما باید در این تابع با صدا کردن تابع `query` سعی کنید یال رسم شده را پیدا کنید و سپس با صدا کردن تابع `setRoad` یال پیدا شده را خروجی دهید.

توجه داشته باشید که گریدر این سوال adaptive است، به این معنا که ساختار درخت در طول اجرای کد شما تعیین می‌شود و پیش از آن مشخص نشده است.

ارزیاب نمونه

فایل‌های مربوط به سوال را از بخش attachments دانلود کنید. راه حل خود را در فایل game.cpp پیاده سازی کنید. برای کامپایل و اجرا کردن این برنامه کامند زیر را درون ترمینال خود اجرا کنید:

```
g++ -o game sample_grader.cpp game.cpp
```

سپس برای اجرا کردن برنامه ی خود فایل اجرایی game که تولید میشود را اجرا کنید (در ویندوز اسم این فایل game.exe و در لینوکس game است)

زیرمسئله‌ها

محدودیت‌ها	نمره	زیرمسئله
$n = 15, m \leq 1500$	7	۱
$n = 50, m \leq 2500$	11	۲
$n = 100, m \leq 2250$	22	۳
$n = 100, m \leq 2000$	21	۴
$n = 100, m \leq 1775$	29	۵
$n = 100, m \leq 1625$	10	۶

ورودی و خروجی نمونه

پرسش/پاسخ شرکت کننده	پاسخ ارزیاب
-	run(4)
query(1, 3, {1}, {2, 3, 4})	return 0
query(1, 2, {2}, {3, 4})	return 1
query(1, 1, {2}, {3})	return 0
setRoad(2, 4)	-
query(2, 2, {2, 4}, {1, 3})	return 0
setRoad(1, 3)	-
query(2, 2, {2, 4}, {1, 3})	return 1
query(1, 2, {2}, {1, 3})	return 0
query(1, 1, {4}, {3})	return 0
setRoad(4, 1)	exit