



به نام خداوند بخشنده مهربان



ساختمان داده - تمرین کامپیوتری چهارم

تاریخ تحویل ۲۰ خرداد ۹۷

گراف دوبخشی

نیما به مسائل گراف بسیار علاقه‌مند است. او به تازگی به مساله‌ای برخورد کرده که ذهنش را مشغول کرده است: اگر یک درخت با n راس داشته باشیم، و بخواهیم به آن یال اضافه کنیم، به شرطی که گراف حاصل همچنان دوبخشی بماند، حداکثر تعداد یال‌هایی که می‌توانیم اضافه کنیم که این شرط دوبخشی بودن گراف برقرار بماند چند است؟

* در مورد گراف دوبخشی می‌توانید اینجا را مطالعه کنید

ورودی

در خط اول ورودی یک عدد $1 \leq n \leq 10^5$ می‌آید که تعداد راس‌های درخت اولیه است. در $n-1$ خط بعدی، در هر خط دو عدد می‌آید که یال‌هایی است که دو راس را بهم متصل می‌کند. (یال‌ها از ۱ تا n شماره گذاری شده‌اند) تضمین می‌شود که گراف ورودی داده‌شده یک درخت است.

خروجی

در تنها خط خروجی یک عدد چاپ کنید که نشان‌دهنده‌ی حداکثر تعداد یال‌هایی است که می‌توان به درخت ورودی اضافه کرد که گراف همچنان دوبخشی بماند.

Input:

3

1 2

1 3

Output:

0

Input:

5

1 2

2 3

3 4

4 5

Output:

2

توضیح

در نمونه اول درخت تولید شده یک گراف دوبخشی کامل است، در نتیجه هیچ یال دیگری نمی‌توانیم به آن اضافه کنیم.

در نمونه‌ی دوم می‌توانیم یال‌های (۱ و ۴) و (۲ و ۵) را اضافه کنیم.

اتوبوس یا قطار؟

نیما در شهری زندگی می‌کند که جاده‌های بین محله‌های آن از یک قانون بسیار ساده پیروی می‌کنند. در این شهر، محله‌ها یا فقط از طریق یک خیابان دوطرفه به هم متصلند و یا فقط از طریق یک ریل دوطرفه قطار. به عبارت دیگر بین هر دو شهر ریل قطار است اگر و فقط اگر خیابان نباشد. رفتن از محله‌ای به محله‌ی دیگر در صورتی که به صورت مستقیم با ریل یا با خیابان به هم متصل باشند، دقیقاً یک ساعت طول می‌کشد. این شهر n محله دارد که از ۱ تا n شماره گذاری شده‌اند. نیما می‌خواهد بداند اگر یک اتوبوس و یک قطار هم‌زمان از محله‌ی ۱ شروع به حرکت کنند، حداقل زمان لازم برای اینکه هر دو به شهر n رسیده باشند؟

* توجه کنید که بجز در شهر n ، در هیچ شهر دیگری قطار و اتوبوس نباید به هم برخورد کنند.

ورودی

در خط اول ورودی دو عدد $0 \leq m \leq n(n-1)/2$ ، $0 \leq n \leq 400$ که n تعداد محله‌های شهر و m تعداد ریل‌های شهر است. در هر خط دو عدد آمده که نشان‌دهنده‌ی اتصال دو محله از طریق ریل قطار است.

خروجی

در تنها خط خروجی یک عدد چاپ کنید که نشان‌دهنده‌ی کمترین زمان لازم برای رسیدن هر دوی قطار و اتوبوس به محله‌ی n ام با شروع از محله‌ی ۱ است. در صورتی که یکی از اتوبوس یا قطار نمی‌توانند به محله‌ی n برسند، عدد -1 را چاپ کنید.

Input:

4 2

1 3

3 4

Output:

2

Input:

4 6

1 2

1 3

1 4

2 3

2 4

3 4

Output:

-1

توضیح

در نمونه‌ی اول برای مثال قطار می‌تواند از مسیر $4 > 3 > 1$ و اتوبوس از مسیر $4 > 1$ از محله‌ی ۱ به ۴ بروند. و مسیر اتوبوس ۱ ساعت و مسیر قطار دو ساعت طول می‌کشد، در نتیجه حداقل ۲ ساعت طول می‌کشد که هر دو به شهر ۴ رسیده باشند.

در نمونه دوم هم همه‌ی شهرها فقط از طریق قطار به هم متصلند در نتیجه با اتوبوس اصلاً نمی‌توان به شهر ۴ رسید.

واژه‌نامه عجیب

* توجه کنید که این سوال امتیازی است.

نیما یک واژه‌نامه (dictionary) پیدا کرده است، این واژه‌نامه کمی عجیب است. در این واژه‌نامه n کلمه وجود دارد که همه کلمات مرتب شده اند، اما نه به زبان عادی، به یک زبان بیگانه که حروف الفبای آن از k حرف اول الفبای انگلیسی تشکیل شده است، اما ترتیب آن با ترتیب حروف انگلیسی متفاوت است. او از شما کمک خواسته که ترتیب حروف الفبا در این زبان عجیب را پیدا کنید. شما باید با کلمات موجود در واژه‌نامه، یک رشته به طول k در خروجی برگردانید که ترتیب حروف در این واژه‌نامه‌ی عجیب است.

ورودی

در خط اول ورودی دو عدد n و k داده می‌شود که به ترتیب تعداد کلمات موجود در واژه‌نامه و تعداد حروف الفبای انگلیسی موجود در واژه‌نامه با شروع از حرف اول است. در خط بعدی n کلمه انگلیسی داده می‌شود که فقط شامل k حرف اول الفبای انگلیسی هستند. دقت کنید که این کلمات به ترتیب حروف آن زبان عجیب مرتب شده هستند.

خروجی

یک رشته به طول k که ترتیب حروف الفبای زبان عجیب با توجه به کلمات داده شده است. برای یکسان شدن جواب، در صورتی که بیش از یک حالت جواب برای یک مساله وجود داشت، رشته‌ای را چاپ کنید که از نظر الفبای انگلیسی کوچک‌تر باشد. (حرف a از b کوچکتر است، در نتیجه اگر در جواب a و b به هر ترتیبی می‌توانند قرار بگیرند، a را زودتر از b چاپ کنید).

Input:

5 4

baa abcd abca cab cad

Output:

bdac

Input:

4 3

caa aaa aab baa

Output:

cab

توضیح

برای مثال در نمونه‌ی دوم، کلمه‌ی *caa* از *aaa* زودتر آمده است، پس در الفبای این زبان، حرف *c* قبل از حرف *a* است، به همین ترتیب کلمه‌ی *baa* بعد از کلمه‌ی *aaa* آمده است، پس *b* بعد از *a* قرار دارد، در نتیجه رشته‌ی خروجی به شکل *cab* می‌شود.

این مساله با استفاده از *topological sort* حل می‌شود، شما می‌توانید با مراجعه به این لینک و یا ویکی‌پدیا در مورد این نوع مرتب‌سازی اطلاعات بیشتری بدست آورید.

شاد باشید.