

Zboruri

В определённой стране есть N городов, обозначенных номерами $1, 2, \dots, N$ и соединённых между собой двусторонними авиарейсами. В городе есть диспетчерская, которая управляет расписанием авиарейсов. Каждый день диспетчерская вносит изменения в расписание авиарейсов следующим образом:

- Выбирается город X .
- Добавляются рейсы из города X во все остальные города, куда рейсы не осуществляются.
- Все существующие ранее рейсы из города X отменяются.

Например, если из Города 5 есть рейсы в Города 1 и 2, но нет рейсов в Города 3 и 4, то после внесённых изменений будут рейсы из Города 5 в Города 3 и 4, а рейсы в Города 1 и 2 будут отменены.

Граждане этой страны интересуются, могла ли бы диспетчерская создать полное расписание авиарейсов, то есть расписание, в котором будет прямой рейс между каждой парой различных городов.

Задача. Напишите программу, которая определяет, может ли диспетчерская создать полное расписание авиарейсов, то есть каждая пара городов была бы соединена прямыми рейсами, независимо от последовательности добавления и отмены рейсов.

Входные данные. Первая строка стандартного ввода содержит целое число N – количество городов. Вторая строка содержит целое число M , представляющее количество текущих рейсов. Каждая из следующих M строк содержит два различных числа, разделённых пробелом – метки городов, которые в настоящее время соединены.

Выходные данные. На одной строке будет выведен ответ «DA» или «NU», соответственно, может ли диспетчерская создать полное расписание авиарейсов или нет.

Ограничения. $2 \leq N \leq 1\,000$, $0 \leq M < N * (N - 1) / 2$. Ограничения по времени выполнения и объёму используемой памяти даны в общем описании задач, предложенных для решения. Исходный файл будет иметь имя `zboruri.pas`, `zboruri.c` или `zboruri.cpp`.

Пример 1.

Вход

```
2
0
```

Выход

```
DA
```

Объяснение: В данном примере 2 города и ни одного текущего рейса. На следующий день будет прямой рейс между этими двумя городами.

Пример 2.

Вход

```
3
2
1 2
2 3
```

Выход

```
NU
```

Объяснение: В данном примере 3 города и 2 текущих рейса. Можно заметить, что в четные дни будут рейсы 1-2 и 2-3, а в нечетные дни будет рейс 1-3, так что никогда не будет всех рейсов в один и тот же день.

Оценка. Тесты будут сгруппированы в следующие категории:

- Для 15% тестов: $N \leq 15$.
- Для других 23% тестов: $N \leq 25$.
- Для других 53% тестов: $N \leq 500$.
- Для остальных тестов (9%) нет дополнительных ограничений.