

Общее описание задач

№	Название задачи	Ограничение на объем используемой памяти	Ограничение на время выполнения программы, секунды	Количество баллов, присвоенных задаче
1.	Tetris	≤ 256 Мб	$\leq 2,0$	100
2.	Cursuri	≤ 256 Мб	$\leq 2,0$	100
3.	Orașe	≤ 256 Мб	$\leq 2,0$	100

Примечание. При равенстве общего количество очков, приоритет будет отдан участнику, набравшему первым соответствующее количество очков.

Tetris

Видеоигра Тетрис представляет собой прямоугольную сетку с N строками и M столбцами. Перед началом игры некоторые ячейки сетки пусты (обозначены как `.`), а другие заполнены (обозначены как `#`). Заполненные квадраты представляют собой набор объектов, а заполненные квадраты, которые смежны (по горизонтали или вертикали), должны считаться частью одного и того же составного объекта. В приведенном ниже примере представлено четыре составных объекта (окрашены разными цветами).

```

. . # .
# # . #
. # # .
# . . .
# . . .

```

Когда игра начинается, объекты падают прямо вниз на сетку, все с одинаковой скоростью. Каждый объект продолжает падать прямо вниз, пока не достигнет нижней строки или пока его часть не приземлится непосредственно над другим объектом, в этом случае он останавливается.

Ваша задача - определить конечное состояние сетки..

Входные данные. Первая строка стандартного ввода содержит два целых числа N и M - количество строк и столбцов сетки. Следующие N строк содержат по M символов каждая, описывающих начальное состояние сетки. Если столбец j строки i сетки содержит блок, соответствующий символ ввода будет `#`, в противном случае он будет символом `.`.

Выходные данные. Стандартный вывод должен содержать N строк, каждая из которых содержит M символов, описывающих конечное состояние сетки. Если столбец j строки i сетки содержит блок, соответствующий символ ввода будет `#`, в противном случае он будет символом `.`.

Ограничения. $1 \leq N, M \leq 10^6$. Дополнительные ограничения на время выполнения и использование памяти указаны в общем описании предлагаемых задач для решения. Исходный файл должен называться tetris.pas, tetris.c или tetris.cpp.

Пример 1.

Входные данные

```

5 4
. . # .
# # . #
. # # .
# . . .
# . . .

```

Выходные данные

```

. . . .
. . . .
# # # .
# # # .
# . . #

```

Оценивание: Тесты организованы следующим образом:

- Для 31% тестов, $N * M \leq 2000$.
- Для других 15% тестов, $M = 2$.
- Оставшиеся 54% тестов не имеют дополнительных ограничений.

Cursuri

В этом году Технический университет Молдовы ввел внеклассные курсы для N студентов первого курса. Каждый студент должен выбрать один из двух доступных курсов. Условно будем считать, что студент i должен выбрать один из двух доступных курсов x_i и y_i .

Для лучшей организации курсов руководство университета решило группировать студентов по M критериям. Критерии описываются тремя целыми числами: a , b и c . Если a равно 1, то студент b не может посещать тот же курс, что и студент c . Если a равно 2, то студент b будет посещать тот же курс, что и студент c .

Задача. Напишите программу, которая, учитывая курсы, которые может посещать каждый студент, и критерии распределения студентов, определяет распределение, соответствующее всем M критериям.

Входные данные. Первая строка стандартного вывода содержит два целых числа, разделенных пробелом, N и M . Следующие N строк содержат две строки символов, разделенные пробелом. Строка i содержит строки x_i и y_i . Каждая строка не будет содержать более 20 символов. Следующие M строк содержат три целых числа, разделенных пробелом. Строка j содержит числа a_j , b_j , c_j ($a_j = 1$ или 2 , $1 \leq b_j, c_j \leq N$).

Выходные данные. Стандартный вывод будет содержать N строк, где строка i содержит название курса, выбранного студентом i . Если никакое распределение не является возможным, будет напечатано единственное число, -1 .

Ограничения. $1 \leq N, M \leq 3 \cdot 10^5$. Ограничения по времени выполнения и объему используемой памяти даны в общем описании задач, предложенных для решения. Исходный файл должен иметь название `cursuri.pas`, `cursuri.c` или `cursuri.cpp`.

Пример 1.

Вход

```
4 4
informatica chimia
chimia informatica
informatica matematica
matematica informatica
2 1 2
1 1 4
1 3 4
1 1 3
```

Выход

```
chimia
chimia
informatica
matematica
```

Объяснение: Замечаем, что только студенты 1 и 2 могут посещать один и тот же курс вместе. Учитывая, что курс `informatica` может посещать только студент 3 или 4, делаем вывод, что студенты 1 и 2 будут посещать курс `chimia`. Студенты 3 и 4 не могут посещать один и тот же курс, соответственно, одна из возможностей заключается в том, что студент 3 будет посещать `informatica`, а студент 4 - `matematica`.

Оценка: Тесты будут организованы следующим образом:

- Для 5% тестов, $N \leq 3$ и $M \leq 2$;
- Для 11% тестов, $N \leq 10$ и $M \leq 20$;
- Для 18% тестов, $N \leq 20$ и $M \leq 200$;
- Для 25% тестов, $N \leq 1000$ и $M \leq 4000$;
- Для оставшихся 41% тестов, нет других ограничений;

Orașe

В Байтландии есть N городов, пронумерованных от 1 до N . В городе i живут A_i граждан. Между городами есть $N - 1$ дорог, пронумерованных от 2 до N . Дорога j соединяет город j и город P_j , где $P_j < j$. К любому городу может быть подключено не более 36 дорог.

Зимой все дороги будут превращены в односторонние автострады из-за опасных условий вождения. То есть, дорога j станет автострадой, которая либо односторонняя от города j к городу P_j , либо односторонняя от города P_j к городу j . Каждый гражданин хочет отправить открытку каждому другому гражданину. Поскольку автострады являются единственными путями передвижения между городами, гражданин x может отправить открытку гражданину y только в том случае, если возможно путешествовать из города, в котором живет гражданин x , в город, в котором живет гражданин y , используя только автомагистраль.

Задача. Напишите программу, которая определяет максимальное количество открыток, которые могут быть отправлены после превращения всех дорог в автострады.

Входные данные. Первая строка стандартного ввода содержит целое число N , количество городов. Вторая строка содержит N целых чисел A_1, A_2, \dots, A_N , разделенных пробелом – население каждого города. Третья строка содержит $N-1$ целых чисел P_2, P_3, \dots, P_N , разделенных пробелом – связь между городами 2, 3, ..., N .

Выходные данные. Первая и единственная строка стандартного вывода будет содержать максимальное количество открыток, которые могут быть отправлены после превращения всех дорог в автострады.

Ограничения. $2 \leq N \leq 200\,000$; $1 \leq A_i \leq 10\,000$, $i = 1, \dots, N$; $1 \leq P_j \leq j$, $j = 2, \dots, N$. Пусть D - максимальное количество дорог, подключенных к любому городу. Гарантируется, что $D \leq 36$. Ограничения по времени выполнения и объему используемой памяти даны в общем описании задач, предложенных для решения. Исходный файл должен иметь название `orase.pas`, `orase.c` или `orase.cpp`.

Пример 1.

Вход

```
4
3 3 4 1
1 2 1
```

Выход

```
67
```

Объяснение: Один из возможных способов превращения дорог в автострады - сделать дорогу 2 односторонней от города 2 к городу 1, дорогу 3 - односторонней от города 3 к городу 2, а дорогу 4 - односторонней от города 1 к городу 4.

Каждый гражданин из города 3 может отправить 3 открытки гражданам города 3, 3 открытки гражданам города 2, 3 открытки гражданам города 1 и 1 открытку гражданину города 4, всего 40 открыток, отправленных из города 3. Аналогично,

- граждане города 2 отправляют по 6 открыток каждый, всего 18 открыток.
- граждане города 1 отправляют по 3 открытки каждый, всего 9 открыток.
- гражданин города 4 не может отправить ни одной открытки.

Всего отправлено $40 + 18 + 9 = 67$ открыток.

Оценка: Тесты будут организованы следующим образом:

- Для 20% тестов, $N \leq 10$.
- Для других 20% тестов, $N \leq 1\,000$ и $D \leq 10$.
- Для других 20% тестов, $D \leq 18$.
- Для других 20% тестов, будет 37 городов, в которых один город соединен с другими 36 городами, а эти другие 36 городов соединены только с этим одним городом.
- Остальные 20% не имеют дополнительных ограничений.