

## Задача А. Решётки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В этом году Боба перешел в 7-й класс Берляндской школы 007. И однажды к ним в школу пришел Биба Бибович, преподаватель Школы математики и программирования Берляндского университета, чтобы предложить им заниматься олимпиадной математикой и программированием. Число желающих было очень велико, поэтому Биба Бибович решил провести отбор, дав ученикам пару задач по математике и одну задачу по программированию. С математикой Боба справился быстро, а вот с программированием у него возникли проблемы. Поэтому он пишет Вам, как лучшему другу, чтобы Вы помогли ему решить задачу и попасть в Школу математики и программирования.

Задача заключается в следующем: даны четыре строки, первая из которых состоит из одного символа «+», «-», «\*» или «/»; а оставшиеся три только из символов «#». Нужно заменить все «#» на цифры так, чтобы при применении операции из первой строки к первым двум полученным числам результат равнялся третьему числу. При этом числа не могут содержать лидирующих нулей и не могут быть отрицательными.

### Формат входных данных

Набор входных данных содержит четыре строки.

Первая строка состоит из одного символа «+», «-», «\*» или «/» (без кавычек).

Оставшиеся строки  $L_i$  состоят только из символов «#».

Гарантируется, что длина каждой из строк  $L_i$  не больше 18.

### Формат выходных данных

Если ответ существует, то выведите четыре строки: в первой строке «YES», а в оставшихся трёх соответствующие числа.

Иначе выведите две строки. В первой строке «NO», а во второй — минимальное число решёток третьей строки, которое нужно **удалить** или **добавить**, чтобы ответ существовал. Если ни при каком изменении третьей строки ответ не существует, то выведите  $-1$ .

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq L_i \leq 18$	100	—	Каждый тест

## Примеры

стандартный ввод	стандартный вывод
+ # # #	YES 1 1 2
* # ## ##	YES 1 10 10
- # ## #	NO -1
/ # # ##	NO 1

## Задача В. Меткий стрелок

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Александр Селедкин с раннего детства увлекается программированием. Уже к 10 годам он бодро писал код на языке *Гладиолус++*. В 11 классе ему наконец-то удалось победить на крупной олимпиаде «Зеленые бублики». Получив за турнир большой денежный приз, главный герой не придумал ничего лучше, чем начать путешествовать по миру.

За время своего отдыха он успел увидеть собственными глазами Египетские пирамиды, Эйфелеву башню, Великую Китайскую стену и даже подружиться с племенем Чунга-Юнга. Но вскоре Александр соскучился по дому и решил вернуться в родной Томск. На радостях он решил прогуляться по знакомым улицам и случайно наткнулся на небольшое здание, на стене которого красовалась вывеска «Тир». Заглянув в свой кошелек, Селедкин обнаружил в нем  $B$  бублей, оставшихся после поездки, и, недолго думая, решил испытать удачу.

На инструктаже Александру объяснили правила игры. Атракцион представляет собой движущийся конвейер, на котором последовательно расположены мишени. Раз в несколько секунд в окне появляется фигурка. Оказавшись на короткое время в зоне видимости, она тут же заменяется следующей за ней мишенью.

Всего существует  $N$  мишеней, каждая из которых окрашена в один из  $K$  цветов. Для выстрела в цель  $i$ -го цвета необходимо заплатить  $M_i$  бублей. При попадании в  $j$ -ю мишень игрок получает  $P_j$  очков, после чего взрывной волной уничтожаются  $Q_j$  следующих за ней фигурок, за которые, к сожалению, очки не начисляются.

Выясните, какое максимальное количество очков сможет набрать Селедкин, а также номера мишеней, по которым нужно выстрелить Селедкину, чтобы набрать эти очки.

### Формат входных данных

Первая строка содержит целое число  $B$  ( $1 \leq B \leq 10^3$ ) — количество бублей у Селедкина.

Вторая строка содержит целое число  $N$  ( $1 \leq N \leq 10^3$ ) — количество фигурок.

Третья строка содержит целое число  $K$  ( $1 \leq K \leq 10^3$ ) — количество различных цветов, в которые окрашены фигурки.

Четвертая строка содержит  $N$  целых чисел  $P_1, P_2, \dots, P_N$  ( $1 \leq P_i \leq 10^9$ ) — количество очков за  $i$ -ю мишень.

Пятая строка содержит  $N$  целых чисел  $C_1, C_2, \dots, C_N$  ( $1 \leq C_i \leq K$ ) — цвет  $i$  мишени.

Шестая строка содержит  $N$  целых чисел  $Q_1, Q_2, \dots, Q_N$  ( $1 \leq Q_i \leq 10^3$ ) — количество мишеней, которые уничтожаются при попадании в  $i$ -ю мишень.

Седьмая строка содержит  $K$  целых чисел  $M_1, M_2, \dots, M_K$  ( $1 \leq M_i \leq 10^3$ ) — количество бублей, необходимых, чтобы выстрелить в  $i$ -й цвет.

### Формат выходных данных

В первой строке выведите два числа — максимальное количество очков, которое сможет заработать Александр и количество мишеней  $T$ , по которым нужно выстрелить.

В второй строке выведите  $T$  чисел — номера мишеней по возрастанию, по которым должен выстрелить Александр. Если существует несколько решений, выведите лексикографически минимальную последовательность номеров мишеней.

Последовательность  $X$  лексикографически меньше последовательности  $Y$ , если либо  $X$  является префиксом  $Y$  (и  $X \neq Y$ ), либо существует такой индекс  $i$  ( $1 \leq i \leq \min(|X|, |Y|)$ ), что  $X_i < Y_i$  и для всех  $j$  ( $1 \leq j < i$ )  $X_j = Y_j$ . Здесь  $|X|$  обозначает длину последовательности  $X$ .

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq B, N, K \leq 10^3$ $1 \leq P_i \leq 10^9$ $1 \leq C_i \leq K$ $1 \leq Q_i, M_i \leq 10^3$	100	—	Каждый тест

### Пример

стандартный ввод	стандартный вывод
100	50 2
3	1 3
2	
20 20 30	
1 2 1	
1 3 1	
45 37	

## Задача С. В поисках палиндрома

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

У друга Поликарпа завтра день рождения. Недолго думая, Поликарп решил подарить ему палиндром, состоящий из строчных букв латинского алфавита. Поликарп знает, что его друг любит некоторые буквы, в количестве  $K$  штук. Поликарпу очень важен его друг, поэтому он хочет подарить ему самый красивый палиндром, содержащий хотя бы одну его любимую букву. Будем считать, что один палиндром красивее другого, если в нем есть любимая буква, стоящая в алфавите позже, чем все любимые буквы второго палиндрома.

В магазине палиндромов продавец выдал Поликарпу строку  $S$  длиной в  $N$  символов, из которой он может выбрать любую подстроку, являющуюся палиндромом. Гарантируется, что в данной строке содержится хотя бы один палиндром.

Помогите Поликарпу найти самый лучший палиндром. Если в строке  $S$  нет палиндромов, содержащих любимые буквы, то выведите любой палиндром, который является подстрокой  $S$ .

**Определение:** *Палиндром* — строка, одинаково читающаяся в обоих направлениях.

### Формат входных данных

Первая строка содержит два целых числа  $N$  и  $K$  ( $1 \leq N \leq 10^5, 1 \leq K \leq 26$ ) — длину строки  $S$  и количество любимых букв.

Вторая строка содержит строку, состоящую из  $K$  строчных латинских различных букв — любимые буквы друга Поликарпа.

Третья строка содержит строку  $S$ , состоящую из  $N$  строчных латинских букв.

### Формат выходных данных

Выведите одну единственную строку — самый лучший палиндром, который можно получить из строки  $S$ .

Если существует несколько палиндромов, удовлетворяющих условию, то выведите любой из них.

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq N \leq 10^5, 1 \leq K \leq 26$	100	—	Каждый тест

### Примеры

стандартный ввод	стандартный вывод
4 1 g gmgi	gmg
8 3 bcn ianfffnb	nfffn
16 7 codevka cejecgaagdeakvka	akvka

## Задача D. Удаления

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Боря — начинающий разработчик. Он занят созданием очень важного приложения, использующего в своей реализации необычную базу данных. Базе данных предстоит хранить всевозможные строки длины не больше  $N$  в определённом порядке. Каждая строка встречается в базе данных ровно один раз и состоит из заглавных букв латинского алфавита.

Приложение Бориса должно выполнить  $K$  операций одного вида:

- удалить все строки в базе, которые находятся между строками  $A_i$  и  $B_i$ , включительно

Порядок строк в базе данных задаётся следующим образом: раньше идёт та строка, у которой меньше длина. При равенстве длин раньше идёт лексикографически меньшая строка. Строка  $X$  лексикографически меньше, чем строка  $Y$ , если либо  $X$  является префиксом  $Y$  (и  $X \neq Y$ ), либо существует такой индекс  $i$  ( $1 \leq i \leq \min(|X|, |Y|)$ ), что  $X_i < Y_i$  и для всех  $j$  ( $1 \leq j < i$ )  $X_j = Y_j$ . Здесь  $|X|$  обозначает длину строки  $X$ .

Борису нужно удостовериться, что его приложение работает правильно. Он просит Вас написать программу, которая выведет количество строк, которые будут удалены из базы после выполнения всех заданных операций.

### Формат входных данных

Первая строка содержит два целых числа  $N$  и  $K$  ( $1 \leq N \leq 12$ ), ( $1 \leq K \leq 10^5$ ) — максимальная длина строк, хранящихся в базе данных, и заданное количество операций удаления соответственно.

Далее следует  $K$  строк, каждая из которых содержит две разделённых пробелом строки  $A_i$  и  $B_i$  — минимальная и максимальная удаляемая в  $i$ -й операции строка. Каждая из заданных строк состоит из заглавных букв латинского алфавита и имеет длину не больше  $N$ .

### Формат выходных данных

Выведите одно целое число — количество строк, которые будут удалены из базы после выполнения всех заданных операций.

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq N \leq 4, 1 \leq K \leq 10^3$	20	—	Полная
2	$1 \leq N \leq 4, 1 \leq K \leq 10^5$	20	1	Полная
3	$1 \leq N \leq 12, 1 \leq K \leq 10^3$	20	1	Полная
4	$1 \leq N \leq 12, 1 \leq K \leq 10^5$	40	2, 3	Каждый тест

## Примеры

стандартный ввод	стандартный вывод
2 1 Z AB	3
1 3 A F D R R X	24
2 6 AB BC CD DF FG GI IJ JK KL LM MN NO	170

## Замечание

В первом примере единственной операцией будут удалены строки «Z», «AA» и «AB».

## Задача Е. Подземелье Варга

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В королевстве Лорлей царил мир и покой, но не долго это длилось. Люди на территории королевства начали бесследно исчезать. Главнокомандующий поручил расследование пропажи людей своим самым лучшим воинам. В ходе расследования оказалось, что похитителями являются так называемые Инсектоиды — разумная раса насекомых, которые устроили логово неподалеку от места жительства людей. На выполнение поручения отправили лучшего из лучших — Варга.

Воину Варгу необходимо зачистить от Инсектоидов  $N$  узких и длинных подземелий. В каждом из  $N$  подземелий он встречает  $T_i$  ( $1 \leq i \leq N$ ) обитателей. Подземелья настолько узкие, что Варг не может обойти очередного монстра. Подземелья соединены между собой в некотором порядке односторонними порталами. Чтобы перейти к следующему подземелью, Варгу необходимо полностью зачистить текущее. У Варга и обитателей подземелья есть определенный уровень. Изначально уровень Варга равняется 1. Варг может уничтожать Инсектоидов своего уровня и ниже. За каждого убитого врага Варгу добавляется  $\lfloor \frac{P-100}{L} \rfloor$  единиц опыта (здесь  $P$  — уровень противника,  $L$  — уровень Варга, 100 единиц опыта равны 1 уровню Варга).

Инсектоид, уровень которого выше уровня Варга, убивает его. Но у Варга есть  $C$  зелий, которые могут понизить уровень Инсектоида на  $X_i$  ( $1 \leq i \leq C$ ) единиц перед началом схватки (зелья можно использовать в любом порядке, но не больше одного зелья на противника). Если уровень Инсектоида после применения зелья опускается ниже уровня Варга, то у Инсектоида происходит линька, и он восстанавливает свой уровень до уровня Варга.

Вам необходимо проверить сможет ли Варг зачистить все подземелья.

Если Варг может зачистить все подземелья, вычислите максимальный уровень, который он может набрать в итоге. В противном случае определите номер Инсектоида (по порядку, начиная с 1), который убьет Варга.

### Формат входных данных

Первая строка содержит два целых числа  $N$  и  $C$  ( $1 \leq N \leq 200$ ,  $2 \leq C \leq 2000$ ) — количество подземелий и количество зелий у Варга соответственно.

Вторая строка содержит  $C$  целых чисел  $X_i$  — мощность  $i$ -го зелья ( $2 \leq X_i \leq 100$ ,  $1 \leq i \leq C$ ).

Далее следует  $2 \cdot N$  строк.

Нечетные строки содержат целое число  $T_i$  — количество обитателей  $i$ -го подземелья ( $1 \leq i \leq N$ ,  $1 \leq T_i \leq 100$ ).

Четные строки содержат  $T_i$  целых чисел  $A_j$  — уровень  $j$ -го обитателя  $i$ -го подземелья ( $1 \leq A_j \leq 250$ ,  $1 \leq j \leq T_i$ ).

### Формат выходных данных

Выведите «YES» (без кавычек) и максимальный целый уровень, который может набрать Варг, если он может зачистить подземелье.

В противном случае выведите «NO» и номер Инсектоида, который убьет Варга.

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq N \leq 200$ $2 \leq C \leq 10 \cdot N$ $1 \leq X_i \leq 250$ $1 \leq T_i \leq 100$ $1 \leq A_j \leq 250$	100	—	Каждый тест



## Примеры

стандартный ввод	стандартный вывод
4 5 4 2 1 1 3 3 1 1 2 2 2 3 6 1 1 2 5 7 8 4 2 3 3 4	YES 10
3 2 1 1 1 1 2 1 1 3 1 1 1	YES 3

## Задача А. Решётки

Пусть  $L_1, L_2$  и  $L_3$  длины первого, второго и третьего чисел соответственно. Рассмотрим следующие операции:

- операция «+», тогда  $L_3 = \max(L_1, L_2)$  (когда, например, первое число равно  $10^{L_1-1}$ , а второе  $10^{L_2-1}$ ) или  $L_3 = \max(L_1, L_2) + 1$  (когда, например, первое число равно  $10^{L_1} - 1$ , а второе  $10^{L_2} - 1$ ), в противном случае, в зависимости от  $L_3$ , легко вычислить сколько символов третьей строки нужно удалить или добавить для получения корректного ответа.
- операция «-». Так как вычитание является обратной операцией по отношению к сложению, то решение практически повторяет предыдущий пункт (необходимо поменять местами значения первого и третьего числа). Дополнительно нужно обработать случай, когда  $L_1 < L_2$ .
- операция «\*», тогда  $L_3 = L_1 + L_2$  (когда, например, первое число равно  $10^{L_1} - 1$ , а второе  $10^{L_2} - 1$ ) или  $L_3 = L_1 + L_2 - 1$  (когда, например, первое число равно  $10^{L_1-1}$ , а второе  $10^{L_2-1}$ ), в противном случае, в зависимости от  $L_3$ , легко вычислить сколько символов третьей строки нужно удалить или добавить для получения корректного ответа.
- операция «/». Так же как и вычитание по отношению к сложению, деление является обратной по отношению к произведению, поэтому решение практически повторяет предыдущий пункт (необходимо поменять местами значения первого и третьего чисел) и дополнительно обработать случай, когда  $L_1 < L_2$ .

Особое внимание при операциях «\*» и «/» следует уделить случаю, когда длины некоторых строк равны 1, так как в этом случае эти числа могут соответствовать нулям.

## Задача В. Меткий стрелок

Пусть  $F(T, B)$  — максимальное количество очков, которые сможет заработать Селедкин, находясь перед  $T$  мишенью и имея при себе  $B$  бублей. Найдем  $F(T, B)$ .

Возможны два варианта событий. При выстреле в  $T$  мишень  $F(T, B) = F(Q_T + T + 1, B - C_T) + P_T$ , иначе  $F(T, B) = F(T + 1, B)$ , где  $C_T$  и  $Q_T$  — это стоимость выстрела в  $T$  фигурку и количество мишеней, которые будут заблокированы. То есть:  $F(T, B) = \max(F(T + 1, B), F(Q_T + T + 1, B - C_T) + P_T)$ .

Все вычисленные значения необходимо «запоминать» в массив размерности  $N \cdot B$ . Также требуется восстановить ответ. Для этого нужно дополнительно запоминать оптимальный выбор для каждого состояния.

## Задача С. В поисках палиндрома

Заметим, что под определение палиндрома подходят строки из одного символа. Тогда решением задачи будет строка, содержащая один единственный символ — любимая буква, принадлежащая строке  $S$ , которая стоит в алфавите позже или на том же месте (если таких букв несколько), чем все остальные любимые буквы строки  $S$ .

Поэтому достаточно отсортировать строку любимых букв в порядке убывания позиций букв в алфавите и проверить каждую букву: есть ли такая любимая буква в строке  $S$ . Если да, то вывести её и закончить проверку букв. Если ни одна буква не содержится в строке  $S$ , то вывести любой палиндром строки  $S$  (подойдёт и первая буква строки  $S$ ).

## Задача D. Удаления

Заметим, что строки  $A_i$  и  $B_i$  представляют числа в 26-ричной системе счисления. Переведем их в 10-чную. Тогда задача сводится к определению длины объединения отрезков на прямой. Для этого можно воспользоваться алгоритмом Кли. Временная сложность решения зависит от используемой сортировки.

## Задача E. Подземелье Варга

Сортируем массив зелий по возрастанию.

Последовательно проходимся по массиву инсектоидов и проверяем каждого инсектоида:

- Если уровень Варга не меньше уровня инсектоида, то он убивает инсектоида и добавляет себе опыт.
- Если уровень инсектоида выше уровня Варга, то ищем первое попавшееся по списку зелье которое не меньше разницы между уровнем инсектоида и уровнем Варга, делаем перерасчёт уровня инсектоида и повторяем предыдущий пункт.
- Если уровень инсектоида после пересчёта остался выше уровня Варга, или зелье вообще не найдено, то выводим «*NO*» и номер инсектоида.

Если Варг уничтожил всех инсектоидов, то выводим «*YES*» и его уровень.