A. Прокат велосипедов

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Как известно, в теплую погоду многие жители крупных городов пользуются сервисами городского велопроката. Вот и Аркадий сегодня будет добираться от школы до дома, используя городские велосипеды.

Школа и дом находятся на одной прямой улице, кроме того, на той же улице есть n точек, где можно взять велосипед в прокат или сдать его. Первый велопрокат находится в точке x_1 километров вдоль улицы, второй — в точке x_2 и так далее, n-й велопрокат находится в точке x_n. Школа Аркадия находится в точке x_1 (то есть там же, где и первый велопрокат), а дом — в точке x_n (то есть там же, где и n-й велопрокат). Известно, что $x_i < x_{i+1}$ для всех $1 \le i < n$.

Согласно правилам пользования велопроката, Аркадий может брать велосипед в прокат только на ограниченное время, после этого он должен обязательно вернуть его в одной из точек велопроката, однако, он тут же может взять новый велосипед, и отсчет времени пойдет заново. Аркадий может брать не более одного велосипеда в прокат одновременно. Если Аркадий решает взять велосипед в какой-то точке проката, то он сдаёт тот велосипед, на котором он до него доехал, берёт ровно один новый велосипед и продолжает на нём своё движение.

За отведенное время, независимо от выбранного велосипеда, Аркадий успевает проехать не больше k километров вдоль улицы.

Определите, сможет ли Аркадий доехать на велосипедах от школы до дома, и если да, то какое минимальное число раз ему необходимо будет взять велосипед в прокат, включая первый велосипед? Учтите, что Аркадий не намерен сегодня ходить пешком.

Входные данные

В первой строке следуют два целых числа n и k $(2 \le n \le 1\,000,\ 1 \le k \le 100\,000)$ — количество велопрокатов и максимальное расстояние, которое Аркадий может проехать на одном велосипеде.

В следующей строке следует последовательность целых чисел $x_1, x_2, \ldots, x_n$ $(0 \le x_1 < x_2 < \ldots < x_n \le 100\,000)$ — координаты точек, в которых находятся велопрокаты. Гарантируется, что координаты велопрокатов заданы в порядке возрастания.

Выходные данные

Если Аркадий не сможет добраться от школы до дома только на велосипедах, выведите `-1`. В противном случае, выведите минимальное количество велосипедов, которые Аркадию нужно взять в точках проката.

Примеры

входные данные

```
4 4
3 6 8 10
```

выходные данные

```
2
```

входные данные

```
2 9
10 20
```

выходные данные

```
-1
```

входные данные

```
12 3
4 6 7 9 10 11 13 15 17 18 20 21
```

выходные данные

```
6
```

Примечание

В первом примере Аркадий должен взять первый велосипед в первом велопрокате и доехать на нём до второго велопроката. Во втором велопрокате он должен взять новый велосипед, на котором он сможет добраться до четвертого велопроката, рядом с которым и находится его дом. Поэтому Аркадию нужно всего два велосипеда, чтобы добраться от школы до дома.

Во втором примере всего два велопроката, расстояние между которыми 10. Но максимальное расстояние, которое можно проехать на одном велосипеде, равно 9. Поэтому Аркадий не сможет добраться от школы до дома только на велосипедах.

Statement is not available on English language

B. Места в самолёте

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

В самолёте есть n рядов мест. Если смотреть на ряды сверху, то в каждом ряду есть 3 места слева, затем проход между рядами, затем 4 центральных места, затем ещё один проход между рядами, а затем ещё 3 места справа.

Известно, что некоторые места уже заняты пассажирами. Всего есть два вида пассажиров — статусные (те, которые часто летают) и обычные.

Перед вами стоит задача рассадить ещё k **обычных** пассажиров так, чтобы суммарное число соседей у статусных пассажиров было минимально возможным. Два пассажира считаются соседями, если они сидят в одном ряду и между ними нет других мест и прохода между рядами. Если пассажир является соседним пассажиром для двух статусных пассажиров, то его следует учитывать в сумме соседей дважды.

Входные данные

В первой строке следуют два целых числа n и k ($1 \le n \le 100$, $1 \le k \le 10 \cdot n$) — количество рядов мест в самолёте и количество пассажиров, которых нужно рассадить.

Далее следует описание рядов мест самолёта по одному ряду в строке. Если очередной символ равен '-', то это проход между рядами. Если очередной символ равен '.', то это свободное место. Если очередной символ равен 'S', то на текущем месте будет сидеть статусный пассажир. Если очередной символ равен 'P', то на текущем месте будет сидеть обычный пассажир.

Гарантируется, что количество свободных мест не меньше k. Гарантируется, что все ряды удовлетворяют описанному в условии формату.

Выходные данные

В первую строку выведите минимальное суммарное число соседей у статусных пассажиров.

Далее выведите план рассадки пассажиров, который минимизирует суммарное количество соседей у статусных пассажиров, в том же формате, что и во входных данных. Если в свободное место нужно посадить одного из k пассажиров, выведите строчную букву 'x' вместо символа '.'.

Примеры

входные данные

```
1 2
SP.-SS.S-S.S
```

выходные данные

```
5
SPx-SSxS-S.S
```

входные данные

```
4 9
PP.-PPPS-S.S
PSP-PPSP-.S.
.S.-S..P-SS.
P.S-P.PP-PSP
```

выходные данные

```
15
PPx-PPPS-S.S
PSP-PPSP-xSx
xSx-SxxP-SSx
P.S-PxPP-PSP
```

Примечание

В первом примере нужно посадить ещё двух обычных пассажиров. Для минимизации соседей у статусных пассажиров, нужно посадить первого из них на третье слева место, а второго на любое из оставшихся двух мест, так как независимо от выбора места он станет соседом двух статусных пассажиров.

Изначально, у статусного пассажира, который сидит на самом левом месте уже есть сосед. Также на четвёртом и пятом местах слева сидят статусные пассажиры, являющиеся соседями друг для друга (что добавляет к сумме 2).

Таким образом, после посадки ещё двух обычных пассажиров, итоговое суммарное количество соседей у статусных пассажиров станет равно пяти.

C. Красивая команда

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Завтра у хоккейной команды, которой руководит Евгений, важный матч. Евгению нужно выбрать шесть игроков, которые выйдут на лед в стартовом составе: один вратарь, два защитника и три нападающих.

Так как это стартовый состав, Евгения больше волнует, насколько красива будет команда на льду, чем способности игроков. А именно, Евгений хочет выбрать такой стартовый состав, чтобы номера любых двух игроков из стартового состава отличались не более, чем в два раза. Например, игроки с номерами 13, 14, 10, 18, 15 и 20 устроят Евгения, а если, например, на лед выйдут игроки с номерами 8 и 17, то это не устроит Евгения.

Про каждого из игроков вам известно, на какой позиции он играет (вратарь, защитник или нападающий), а также его номер. В хоккее номера игроков не обязательно идут подряд. Посчитайте число различных стартовых составов из одного вратаря, двух защитников и трех нападающих, которые может выбрать Евгений, чтобы выполнялось его условие красоты.

Входные данные

Первая строка содержит три целых числа g, d и f ($1 \le g \le 1\,000$, $1 \le d \le 1\,000$, $1 \le f \le 1\,000$) — число вратарей, защитников и нападающих в команде Евгения.

Вторая строка содержит g целых чисел, каждое в пределах от 1 до $100\,000$ — номера вратарей.

Третья строка содержит d целых чисел, каждое в пределах от 1 до $100\,000$ — номера защитников.

Четвертая строка содержит f целых чисел, каждое в пределах от 1 до $100\,000$ — номера нападающих.

Гарантируется, что общее количество игроков не превосходит $1\,000$, т. е. $g + d + f \le 1\,000$. Все $g + d + f$ номеров игроков различны.

Выходные данные

Выведите одно целое число — количество возможных стартовых составов.

Примеры

входные данные

```
1 2 3
15
10 19
20 11 13
```

выходные данные

```
1
```

входные данные

```
2 3 4
16 40
20 12 19
13 21 11 10
```

выходные данные

```
6
```

Примечание

В первом примере всего один вариант для выбора состава, который удовлетворяет описанным условиям, поэтому ответ 1.

Во втором примере подходят следующие игровые сочетания (в порядке вратарь-защитник-защитник-нападающий-нападающий-нападающий):

- `16 20 12 13 21 11`
- `16 20 12 13 11 10`
- `16 20 19 13 21 11`
- `16 20 19 13 11 10`
- `16 12 19 13 21 11`
- `16 12 19 13 11 10`

Таким образом, ответ на этот пример — 6.

Statement is not available on English language

D. Пограничные врата

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Герой Аркадий находится на узкой полоске земли, разделенной на n зон, пронумерованных от 1 до n. Из i-й зоны можно пройти лишь в $(i - 1)$-ю зону и в $(i + 1)$-ю зону, если они существуют. При этом между каждой парой соседних зон находятся пограничные врата, которые могут быть разных цветов, цвет врат между i-й и $(i + 1)$-й зоной равен g_i.

Аркадий может пройти пограничные врата некоторого цвета, только если он перед этим побывал в одном из шатров хранителей ключей этого цвета и взял ключ. В каждой зоне находится ровно один шатер хранителя ключей некоторого цвета, цвет шатра в i-й зоне равен k_i. После посещения шатра определенного цвета Аркадий может неограниченное число раз проходить через любые врата этого цвета.

На проход через одни врата Аркадий тратит один ход, на посещение шатра и другие перемещения ходы не требуются. За какое минимальное число ходов Аркадий может попасть из зоны a в зону b, если изначально у него нет никаких ключей?

Входные данные

Первая строка содержит три целых числа n, a, b ($2 \le n \le 100\,000$, $1 \le a, b \le n$, $a \ne b$) — число зон, номер начальной зоны и номер конечной зоны, соответственно.

Вторая строка содержит $n - 1$ целое число $g_1, g_2, \dots, g_{n-1}$ ($1 \le g_i \le 100\,000$), где g_i означает цвет пограничных врат между зонами i и $i + 1$.

Третья строка содержит n целых чисел $k_1, k_2, \dots, k_n$ ($1 \le k_i \le 100\,000$), где k_i означает цвет шатра хранителя ключей в i-й зоне.

Выходные данные

Если Аркадий не может попасть из зоны a в зону b, не имея изначально ключей, выведите `-1`.

Иначе выведите минимальное количество ходов, которое потребуется Аркадию.

Примеры

входные данные

```
5 4 1
3 1 1 2
7 1 2 1 3
```

выходные данные

```
7
```

входные данные

```
5 1 5
4 3 2 1
4 3 2 5 5
```

выходные данные

```
-1
```

Примечание

В первом примере, чтобы попасть из зоны 4 в зону 1, Аркадию нужно сначала взять ключ цвета 1, пройти в зону 3, там взять ключ цвета 2 и пройти обратно в зону 4 и затем в зону 5, взять там ключ цвета 3 и дойти до зоны 1 за четыре хода.

Во втором примере Аркадий может дойти лишь до четвертой зоны, так как шатров хранителей ключей цвета 1 нет совсем.