

В. 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 ещё один проход между рядами, а затем ещё 3 места справа. 4 центральных места, затем

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 ($1 \leq n \leq 100$, $1 \leq k \leq 10 \cdot n$) — 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.