**Четырехзначный палиндром**

*(Время: 1 сек. Память: 16 Мб Сложность: 10%)*

Требуется написать программу, определяющую, является ли четырехзначное натуральное число N палиндромом, т.е. числом, которое одинаково читается слева направо и справа налево.

**Входные данные**

Входной файл INPUT.TXT содержит натуральное число N (1000 ≤ N ≤ 9999).

**Выходные данные**

В выходной файл OUTPUT.TXT следует вывести слово «YES», если число N является палиндромом, или «NO» – если нет.

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 6116 | YES |
| 2 | 1231 | NO |

**Если холодно снаружи, ты оденься потеплей…**

**Автобусная экскурсия**

*(Время: 1 сек. Память: 16 Мб Сложность: 14%)*

Оргкомитет Московской городской олимпиады решил организовать обзорную экскурсию по Москве для участников олимпиады. Для этого был заказан двухэтажный автобус (участников олимпиады достаточно много и в обычный они не умещаются) высотой 437 сантиметров. На экскурсионном маршруте встречаются N мостов. Жюри и оргкомитет олимпиады очень обеспокоены тем, что высокий двухэтажный автобус может не проехать под одним из них. Им удалось выяснить точную высоту каждого из мостов. Автобус может проехать под мостом тогда и только тогда, когда высота моста превосходит высоту автобуса.

Помогите организаторам узнать, закончится ли экскурсия благополучно, а если нет, то установить, где произойдет авария.

**Входные данные**

Во входном файле INPUT.TXT сначала содержится число N (1 ≤ N ≤ 1000). Далее идут N натуральных чисел, не превосходящих 10000 - высоты мостов в сантиметрах в том порядке, в котором они встречаются на пути автобуса.

**Выходные данные**

В единственную строку выходного файла OUTPUT.TXT нужно вывести фразу "No crash", если экскурсия закончится благополучно. Если же произойдет авария, то нужно вывести сообщение "Crash k", где k - номер моста, где произойдет авария. Фразы выводить без кавычек ровно с одним пробелом внутри.

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 1 763 | No crash |
| 2 | 3 763 245 113 | Crash 2 |
| 3 | 1 437 | Crash 1 |

**Но а как зайдешь в Ойна,**

**Сложи все вещи в…**

**Манхэттенские улицы**

*(Время: 1 сек. Память: 16 Мб Сложность: 17%)*

Система улиц Нью-Йоркского района Манхеттен весьма интересна. В Манхеттене есть n улиц, идущие с запада на восток (авеню), и m улиц, идущие с севера на юг (просто улицы). Ширина каждого авеню и каждой улицы равна d метров, а длина – k метров. При этом каждая улица пересекает каждый авеню и не имеет общих точек с другими улицами, а каждый авеню пересекает каждую улицу и не имеет общих точек с другими авеню.

Разумеется, все авеню и улицы имеют асфальтовое покрытие. Дорожно-ремонтные службы интересуются: сколько квадратных метров асфальта уложено на все авеню и улицы. На перекрестках, без сомнения, асфальт уложен в один слой.

Напишите программу, вычисляющую ответ на их вопрос.

**Входные данные**

Входной файл INPUT.TXT содержит четыре натуральных числа n, m, d, k (1 ≤ n, m, d, k ≤ 109, k > m∙d, k > n∙d).

**Выходные данные**

В выходной файл OUTPUT.TXT выведите ответ на задачу.

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 1 1 5 10 | 75 |
| 2 | 100 10 23 4560 | 11007800 |

**Хмм, но именно в какой?**

**Кстати Ма**l**ьчики с дн**е**м Защит**n**ика отечества «7 мая»**

**Вагоны**

*(Время: 1 сек. Память: 16 Мб Сложность: 23%)*

Ежедневно диспетчеру железнодорожной станции приходится переставлять вагоны во многих поездах, чтобы они шли в заданном порядке. Для этого диспетчер может расцепить пришедший на станцию состав в произвольных местах и переставить образовавшиеся сцепки из одного или нескольких вагонов в произвольном порядке. Порядок вагонов в одной сцепке менять нельзя, также нельзя развернуть всю сцепку так, чтобы последний вагон в сцепке оказался первым в ней.

Диспетчер просит вашей помощи в определении того, какое минимальное число соединений между вагонами необходимо расцепить, чтобы переставить вагоны в составе в требуемом порядке.

**Входные данные**

В первой строке входного файла INPUT.TXT содержится целое число N (1 ≤ N ≤ 100). Во второй строке содержится перестановка натуральных чисел от 1 до N (то есть все натуральные числа от 1 до N в некотором порядке). Числа разделяются одним пробелом. Эта перестановка задает номера вагонов в приходящем на станцию составе. Требуется, чтобы в уходящем со станции составе вагоны шли в порядке их номеров.

**Выходные данные**

Программа должна записать в выходной файл OUTPUT.TXT единственное целое число, равное минимальному количеству соединений между вагонами, которое нужно расцепить в данном составе, чтобы их можно было переставить по порядку.

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 4 3 1 2 4 | 2 |
| 2 | 5 5 4 3 2 1 | 4 |
| 3 | 2 1 2 | 0 |