

Задача А. Конфеты Пети

Имя входного файла: `combination.in`
Имя выходного файла: `combination.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

У Пети было k конфет, и он захотел их раздать ученикам своей параллели. Однако заметил, что конфет у него меньше чем учеников в параллели. Петя сел на скамейку и задумался. Просидев полчаса и доев последнюю конфету он подумал — интересно, а сколько было способов раздать все k конфет n ученикам параллели С, если конфеты нельзя делить, а каждому школьнику можно дать не более одной конфеты.

Формат входного файла

В единственной строке записаны числа n, k ($1 \leq k \leq n \leq 64$).

Формат выходного файла

Выведите единственное число — ответ на задачу.

Примеры

<code>combination.in</code>	<code>combination.out</code>
5 3	10

Задача В. Карточки

Имя входного файла: `cards.in`
Имя выходного файла: `cards.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

На день рождения Пете подарили набор карточек с буквами. Теперь Петя с большим интересом составляет из них разные слова. И вот, однажды, составив очередное слово, Петя заинтересовала вопросом: «А сколько различных слов можно составить из тех же карточек, что и данное?». Помогите ему ответить на этот вопрос.

Формат входного файла

Вводится слово, составленное Петей — строка из маленьких латинских букв не длиннее 15 символов.

Формат выходного файла

Выведите одно целое число — искомое количество слов.

Примеры

<code>cards.in</code>	<code>cards.out</code>
<code>solo</code>	12

Задача С. Перемешиватель

Имя входного файла: `permutator.in`
Имя выходного файла: `permutator.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Пете надоело перемешивать карты для настольных игр перед каждой игрой, и он заказал по интернету специальный автоматический перемешиватель. Перемешиватель принимает на вход стопку из n карт и возвращает ее в перемешанном виде.

Вскоре друзья Пети заметили, что перемешиватель меняет порядок карт всегда одинаково. Карта, стоящая на месте i после перемешивания оказывается на месте p_i . Поэтому было решено пропускать карты через перемешиватель k раз, чтобы никто из игроков не мог проследить, какая карта попала на какое место.

Однако Петя решил, что особенностью аппарата надо воспользоваться и хочет написать программу, которая посчитает, на каком месте окажется каждая карта после k перемешиваний.

Формат входного файла

Первая строка входного файла содержит целые числа n и k ($1 \leq n \leq 1000$, $1 \leq k \leq 1000$). Следующая строка содержит n чисел p_i ($1 \leq p_i \leq n$, все p_i различны).

Формат выходного файла

Выведите в выходной файл n чисел. i -е число должно быть равно финальной позиции карты, которая была в колоде на i месте до перемешивания.

Примеры

<code>permutator.in</code>	<code>permutator.out</code>
5 2 2 3 1 5 4	3 1 2 4 5

Задача D. Перемешиватель 2

Имя входного файла: `permutator2.in`
Имя выходного файла: `permutator2.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Друзья Пети заметили, что он слишком часто выигрывает и догадались, что он написал решение предыдущей задачи. Тогда они решили увеличить ограничение на число перемешиваний до 10^{18} , чтобы программа Пети не успевала вычислять ответ.

Помогите Пете решить задачу для больших значений k .

Формат входных и выходных данных тот же, что и в предыдущей задаче. Пример тоже можно посмотреть там. Единственное отличие этой задачи — ограничение на $k \leq 10^{18}$.

Осталось много свободного места на странице, вот вам еще поняшка :-)



Задача Е. Перемешиватель 3

Имя входного файла: `permutator3.in`
Имя выходного файла: `permutator3.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Друзья Пети поняли, что он опять мухлюет с перемешивателем и решили его проучить. Они хотят подобрать такое k , что после запуска перемешивателя k раз, карты встают на те же места, на которых были до перемешивания. Помогите друзьям найти минимальное такое k .

Формат входного файла

Первая строка входного файла содержит целое число n ($1 \leq n \leq 1000$). Следующая строка содержит n чисел p_i — перестановку, которую производит перемешиватель.

Формат выходного файла

Выведите в выходной файл минимальное число раз, которое нужно запустить перемешиватель, чтобы все карты встали на исходные места. Гарантируется, что ответ не больше 10^{18} .

Примеры

<code>permutator3.in</code>	<code>permutator3.out</code>
5 2 3 1 5 4	6