A. Генератор скобок

Рита по поручению Тимофея наводит порядок в правильных скобочных последовательностях (ПСП), состоящих только из круглых скобок (). Для этого ей надо сгенерировать все ПСП длины *2n* в алфавитном порядке —– алфавит состоит из ( и ) и открывающая скобка идёт раньше закрывающей.

Помогите Рите —– напишите программу, которая по заданному *n* выведет все ПСП в нужном порядке.

Рассмотрим второй пример. Надо вывести ПСП из четырёх символов. Таких всего две:

1. (())
2. ()()

(()) идёт раньше ()(), так как первый символ у них одинаковый, а на второй позиции у первой ПСП стоит (, который идёт раньше ).

Формат ввода

На вход функция принимает *n* — целое число от *0* до *10*.

Формат вывода

Функция должна напечатать все возможные скобочные последовательности заданной длины в алфавитном (лексикографическом) порядке.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3 | ((()))  (()())  (())()  ()(())  ()()() |

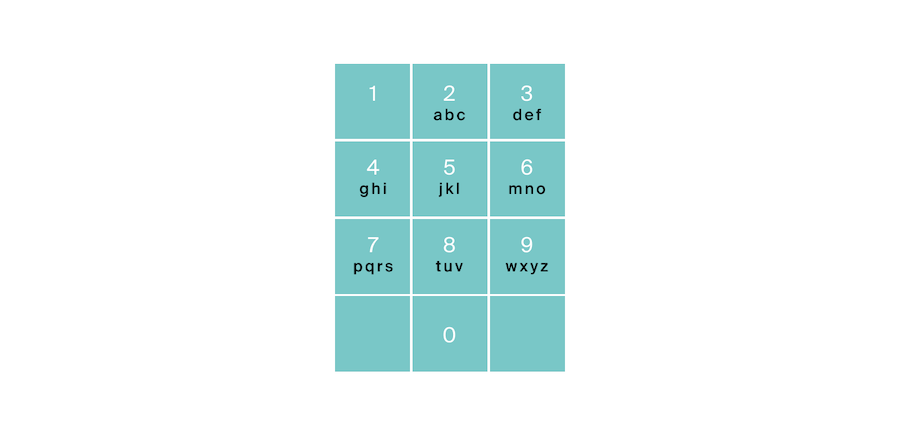
Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2 | (())  ()() |

# B. Комбинации

На клавиатуре старых мобильных телефонов каждой цифре соответствовало несколько букв. Примерно так:

2:'abc',  
3:'def',  
4:'ghi',  
5:'jkl',  
6:'mno',  
7:'pqrs',  
8:'tuv',  
9:'wxyz'

Вам известно в каком порядке были нажаты кнопки телефона, без учета повторов. Напечатайте все комбинации букв, которые можно набрать такой последовательностью нажатий.

## Формат ввода

На вход подается строка, состоящая из цифр 2-9 включительно. Длина строки не превосходит *10* символов.

## Формат вывода

Выведите все возможные комбинации букв через пробел.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 23 | ad ae af bd be bf cd ce cf |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 92 | wa wb wc xa xb xc ya yb yc za zb zc |

# C. Подпоследовательность

Гоша любит играть в игру «Подпоследовательность»: даны 2 строки, и нужно понять, является ли первая из них подпоследовательностью второй. Когда строки достаточно длинные, очень трудно получить ответ на этот вопрос, просто посмотрев на них. Помогите Гоше написать функцию, которая решает эту задачу.

## Формат ввода

В первой строке записана строка s.

Во второй —- строка t.

Обе строки состоят из маленьких латинских букв, длины строк не превосходят 150000. Строки могут быть пустыми.

## Формат вывода

Выведите True, если s является подпоследовательностью t, иначе —– False.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| abf  ahbgdcu | True |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| abcp  ahpc | False |

# D. Печеньки

К Васе в гости пришли одноклассники. Его мама решила угостить ребят печеньем.

Но не всё так просто. Печенья могут быть разного размера. А у каждого ребёнка есть фактор жадности —– минимальный размер печенья, которое он возьмёт. Нужно выяснить, сколько ребят останутся довольными в лучшем случае, когда они действуют оптимально.

Каждый ребёнок может взять не больше одного печенья.

## Формат ввода

В первой строке записано *n* —– количество детей.

Во второй —– *n* чисел, разделённых пробелом, каждое из которых –— фактор жадности ребёнка. Это натуральные числа, не превосходящие *1000*.

В следующей строке записано число *m* –— количество печенек.

Далее —– *m* натуральных чисел, разделённых пробелом —– размеры печенек. Размеры печенек не превосходят *1000*.

Оба числа *n* и *m* не превосходят *10000*.

## Формат вывода

Нужно вывести одно число –— количество детей, которые останутся довольными

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2  1 2  3  2 1 3 | 2 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  2 1 3  2  1 1 | 1 |

# E. Покупка домов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Язык | Ограничение времени | Ограничение памяти | Ввод | Вывод |
| Все языки | 0.05 секунд | 64Mb | стандартный ввод или input.txt | стандартный вывод или output.txt |
| Node.js 14.15.5 | 0.085 секунд | 64Mb |
| Mono C# 5.2.0 | 0.2 секунды | 64Mb |
| Oracle Java 8 | 0.2 секунды | 64Mb |
| OpenJDK Java 11 | 0.2 секунды | 64Mb |
| Node JS 8.16 | 0.085 секунд | 64Mb |

Тимофей решил купить несколько домов на знаменитом среди разработчиков Алгосском архипелаге. Он нашёл *n* объявлений о продаже, где указана стоимость каждого дома в алгосских франках. А у Тимофея есть *k* франков. Помогите ему определить, какое наибольшее количество домов на Алгосах он сможет приобрести за эти деньги.

## Формат ввода

В первой строке через пробел записаны натуральные числа *n* и *k*.

*n* — количество домов, которые рассматривает Тимофей, оно не превосходит *1000*;

*k* — общий бюджет, не превосходит *10000*;

В следующей строке через пробел записано *n* стоимостей домов. Каждое из чисел не превосходит *10000*. Все стоимости — натуральные числа.

## Формат вывода

Выведите одно число —– наибольшее количество домов, которое может купить Тимофей.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3 300  999 999 999 | 0 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3 1000  350 999 200 | 2 |

# 

# F. Периметр треугольника

|  | Все языки | Python 3.7.3 | GNU c++17 7.3 |
| --- | --- | --- | --- |
| Ограничение времени | 1 секунда | 0.12 секунд | 0.015 секунд |
| Ограничение памяти | 64Mb | 64Mb | 64Mb |
| Ввод | стандартный ввод или input.txt | | |
| Вывод | стандартный вывод или output.txt | | |

Перед сном Рита решила поиграть в игру на телефоне. Дан массив целых чисел, в котором каждый элемент обозначает длину стороны треугольника. Нужно определить максимально возможный периметр треугольника, составленного из сторон с длинами из заданного массива. Помогите Рите скорее закончить игру и пойти спать.

Напомним, что из трёх отрезков с длинами *a ≤ b ≤ c* можно составить треугольник, если выполнено **неравенство треугольника**: *c < a + b*

Разберём пример:  
даны длины сторон *6, 3, 3, 2*. Попробуем в качестве наибольшей стороны выбрать *6*. Неравенство треугольника не может выполниться, так как остались *3, 3, 2* —– максимальная сумма из них равна *6*.

Без шестёрки оставшиеся три отрезка уже образуют треугольник со сторонами *3, 3, 2*. Неравенство выполняется: *3 < 3 + 2*. Периметр равен *3 + 3 + 2 = 8*.

## Формат ввода

В первой строке записано количество отрезков *n*, *3≤ n≤ 10000*.

Во второй строке записано *n* натуральных чисел, не превосходящих *10 000*, –— длины отрезков.

## Формат вывода

Нужно вывести одно число —– наибольший периметр треугольника.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  6 3 3 2 | 8 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  5 3 7 2 8 3 | 20 |

# 

# G. Гардероб

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 128Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Рита решила оставить у себя одежду только трёх цветов: розового, жёлтого и малинового. После того как вещи других расцветок были убраны, Рита захотела отсортировать свой новый гардероб по цветам. Сначала должны идти вещи розового цвета, потом —– жёлтого, и в конце —– малинового. Помогите Рите справиться с этой задачей.

Примечание: попробуйте решить задачу за один проход по массиву!

## Формат ввода

В первой строке задано количество предметов в гардеробе: *n* –— оно не превосходит *1000000*. Во второй строке даётся массив, в котором указан цвет для каждого предмета. Розовый цвет обозначен *0*, жёлтый —– *1*, малиновый –— *2*.

## Формат вывода

Нужно вывести в строку через пробел цвета предметов в правильном порядке.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 7  0 2 1 2 0 0 1 | 0 0 0 1 1 2 2 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5  2 1 2 0 1 | 0 1 1 2 2 |

### Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  2 1 1 2 0 2 | 0 1 1 2 2 2 |

# 

# H. Большое число

Вечером ребята решили поиграть в игру «Большое число».  
Даны числа. Нужно определить, какое самое большое число можно из них составить.

## Формат ввода

В первой строке записано *n* — количество чисел. Оно не превосходит *100*.  
Во второй строке через пробел записаны *n* неотрицательных чисел, каждое из которых не превосходит *1000*.

## Формат вывода

Нужно вывести самое большое число, которое можно составить из данных чисел.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  15 56 2 | 56215 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  1 783 2 | 78321 |

### Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5  2 4 5 2 10 | 542210 |

# I. Любители конференций

На IT-конференции присутствовали студенты из разных вузов со всей страны. Для каждого студента известен ID университета, в котором он учится.

Тимофей предложил Рите выяснить, из каких k вузов на конференцию пришло больше всего учащихся.

## Формат ввода

В первой строке дано количество студентов в списке —– *n* (*1 ≤ n ≤ 15 000*).

Во второй строке через пробел записаны *n* целых чисел —– ID вуза каждого студента. Каждое из чисел находится в диапазоне от *0* до *10 000*.

В третьей строке записано одно число *k*.

## Формат вывода

Выведите через пробел *k* ID вузов с максимальным числом участников. Они должны быть отсортированы по убыванию популярности (по количеству гостей от конкретного вуза). Если более одного вуза имеет одно и то же количество учащихся, то выводить их ID нужно в порядке возрастания.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 7  1 2 3 1 2 3 4  3 | 1 2 3 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  1 1 1 2 2 3  1 | 1 |

J. Пузырёк

Чтобы выбрать самый лучший алгоритм для решения задачи, Гоша продолжил изучать разные сортировки. На очереди сортировка пузырьком — [https://ru.wikipedia.org/wiki/Сортировка\_пузырьком](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_%D0%BF%D1%83%D0%B7%D1%8B%D1%80%D1%8C%D0%BA%D0%BE%D0%BC)

Её алгоритм следующий (сортируем по неубыванию):

1. На каждой итерации проходим по массиву, поочередно сравнивая пары соседних элементов. Если элемент на позиции i больше элемента на позиции i + 1, меняем их местами. После первой итерации самый большой элемент всплывёт в конце массива.
2. Проходим по массиву, выполняя указанные действия до тех пор, пока на очередной итерации не окажется, что обмены больше не нужны, то есть массив уже отсортирован.
3. После не более чем n – 1 итераций выполнение алгоритма заканчивается, так как на каждой итерации хотя бы один элемент оказывается на правильной позиции.

Помогите Гоше написать код алгоритма.

Формат ввода

В первой строке на вход подаётся натуральное число n — длина массива, *2 ≤ n ≤ 1000*.  
Во второй строке через пробел записано *n* целых чисел.  
Каждое из чисел по модулю не превосходит *1000*.  
  
Обратите внимание, что считывать нужно только 2 строки: значение *n* и входной массив.

Формат вывода

После каждого прохода по массиву, на котором какие-то элементы меняются местами, выводите его промежуточное состояние.  
Таким образом, если сортировка завершена за *k* меняющих массив итераций, то надо вывести *k* строк по *n* чисел в каждой — элементы массива после каждой из итераций.

Если массив был изначально отсортирован, то просто выведите его.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5  4 3 9 2 1 | 3 4 2 1 9  3 2 1 4 9  2 1 3 4 9  1 2 3 4 9 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5  12 8 9 10 11 | 8 9 10 11 12 |

L. Два велосипеда

Вася решил накопить денег на два одинаковых велосипеда — себе и сестре. У Васи есть копилка, в которую каждый день он может добавлять деньги (если, конечно, у него есть такая финансовая возможность). В процессе накопления Вася не вынимает деньги из копилки.

У вас есть информация о росте Васиных накоплений — сколько у Васи в копилке было денег в каждый из дней.

Ваша задача — по заданной стоимости велосипеда определить

* первый день, в которой Вася смог бы купить один велосипед,
* и первый день, в который Вася смог бы купить два велосипеда.

**Подсказка:** решение должно работать за O(log n).

Формат ввода

В первой строке дано число дней n, по которым велись наблюдения за Васиными накоплениями. *1 ≤ n ≤ 106*.

В следующей строке записаны n целых неотрицательных чисел. Числа идут в порядке неубывания. Каждое из чисел не превосходит *106*.

В третьей строке записано целое положительное число *s* — стоимость велосипеда. Это число не превосходит *106*.

Формат вывода

Нужно вывести два числа — номера дней по условию задачи.

Если необходимой суммы в копилке не нашлось, нужно вернуть *-1* вместо номера дня.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  1 2 4 4 6 8  3 | 3 5 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  1 2 4 4 4 4  3 | 3 -1 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  1 2 4 4 4 4  10 | -1 -1 |

# M. Золотая середина

**Задача повышенной сложности**

На каждом острове в архипелаге Алгосы живёт какое-то количество людей или же остров необитаем (тогда на острове живёт *0* людей). Пусть на *i*-м острове численность населения составляет *ai*. Тимофей захотел найти медиану среди всех значений численности населения.

**Определение:** Медиана (https://ru.wikipedia.org/wiki/Медиана\_(статистика)) массива чисел a\_i —– это такое число, что половина чисел из массива не больше него, а другая половина не меньше. В общем случае медиану массива можно найти, отсортировав числа и взяв среднее из них. Если количество чисел чётно, то возьмём в качестве медианы полусумму соседних средних чисел, *(a[n/2] + a[n/2 + 1])/2*.

У Тимофея уже есть отдельно данные по северной части архипелага и по южной, причём значения численности населения в каждой группе отсортированы по неубыванию.

Определите медианную численность населения по всем островам Алгосов.

**Подсказка:** Если *n* –— число островов в северной части архипелага, а *m* –— в южной, то ваше решение должно работать за https://contest.yandex.ru/testsys/tex/render/TyhcbG9nKG4gKyBtKSk=.png.

## Формат ввода

В первой строке записано натуральное число *n*, во второй —– натуральное число *m*. Они не превосходят *10 000*.

Далее в строку через пробел записаны *n* целых неотрицательных чисел, каждое из которых не превосходит *10 000*, –— значения численности населения в северной части Алгосов.

В последней строке через пробел записаны m целых неотрицательных чисел, каждое из которых не превосходит *10 000* –— значения численности населения в южной части Алгосов.

Значения в третьей и четвёртой строках упорядочены по неубыванию.

## Формат вывода

Нужно вывести одной число — найденную медиану.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2  1  1 3  2 | 2 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2  2  1 2  3 4 | 2.5 |

### Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 8  10  0 0 0 1 3 3 5 10  4 4 5 7 7 7 8 9 9 10 | 5 |

# N. Клумбы

Алла захотела, чтобы у неё под окном были узкие клумбы с тюльпанам. На схеме земельного участка клумбы обозначаются просто горизонтальными отрезками, лежащими на одной прямой. Для ландшафтных работ было нанято n садовников. Каждый из них обрабатывал какой-то отрезок на схеме. Процесс был организован не очень хорошо, иногда один и тот же отрезок или его часть могли быть обработаны сразу несколькими садовниками. Таким образом, отрезки, обрабатываемые двумя разными садовниками, сливаются в один. Непрерывный обработанный отрезок затем станет клумбой. Нужно определить границы будущих клумб.

Рассмотрим примеры.

**Пример 1:**  
Два отрезка *[7, 8]* сливаются в один, но потом их накрывает отрезок *[6, 10]*. Таким образом, имеем две клумбы с координатами *[2,3]* и *[6,10]*.

**Пример 2**  
Отрезки *[2,3]*, *[3, 4]* и *[3,4]* сольются в один отрезок *[2,4]*. Отрезок *[5,6]* ни с кем не объединяется, добавляем его в ответ.

## Формат ввода

В первой строке задано количество садовников *n*. Число садовников не превосходит *100 000*.

В следующих *n* строках через пробел записаны координаты клумб в формате: start end, где start —– координата начала, end —– координата конца. Оба числа целые, неотрицательные и не превосходят *107*. start строго меньше, чем end.

## Формат вывода

Нужно вывести координаты каждой из получившихся клумб в отдельных строках. Данные должны выводится в отсортированном порядке —– сначала клумбы с меньшими координатами, затем —– с бОльшими.

### Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  7 8  7 8  2 3  6 10 | 2 3  6 10 |

### Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  2 3  5 6  3 4  3 4 | 2 4  5 6 |

### Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6  1 3  3 5  4 6  5 6  2 4  7 10 |  |

O. Разность треш-индексов

Гоша долго путешествовал и измерил площадь каждого из *n* островов Алгосов, но ему этого мало! Теперь он захотел оценить, насколько разнообразными являются острова в составе архипелага.

Для этого Гоша рассмотрел все пары островов (таких пар, напомним, *n \* (n-1) / 2*) и посчитал попарно разницу площадей между всеми островами. Теперь он собирается упорядочить полученные разницы, чтобы взять *k*-ую по порядку из них.

Помоги Гоше найти *k*-ю минимальную разницу между площадями эффективно.

**Пояснения к примерам**

Пример 1

Выпишем все пары площадей и найдём соответствующие разницы

1. *|2 - 3| = 1*
2. *|3 - 4| = 1*
3. *|2 - 4| = 2*

Так как нам нужна 2-я по величине разница, то ответ будет *1*.

Пример 2

У нас есть два одинаковых элемента в массиве —– две единицы, поэтому минимальная (первая) разница равна нулю.

Формат ввода

В первой строке записано натуральное число *n* –— количество островов в архипелаге (*2 ≤ n ≤ 10 000*).

В следующей строке через пробел записаны n площадей островов — n натуральных чисел, каждое из которых не превосходит *1 000 000*.

В последней строке задано число k. Оно находится в диапазоне от *1* до *n(n - 1) / 2*.

Формат вывода

Выведите одно число –— *k*-ую минимальную разницу.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  2 3 4  2 | 1 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  1 3 1  1 | 0 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  1 3 5  3 | 4 |

P. Частичная сортировка

После того, как Гоша узнал про сортировку слиянием и быструю сортировку, он решил придумать свой метод сортировки, который предполагал бы разделение данных на части.  
Назвал он свою сортировку Частичной.  
Этим методом можно отсортировать *n* уникальных чисел *a1, a2, … , an*, находящихся в диапазоне от *0* до *n - 1*.  
Алгоритм сортировки состоит из трёх шагов:

* Разбить исходную последовательность на *k* блоков *B1, …, Bk*. Блоки могут иметь разные размеры. Если размер блока *i* равен *si*, то *B1 ={ a1, …, as1 }*, *B2 = { as1 + 1, … , as1 + s2 }* и так далее.
* Отсортировать каждый из блоков.
* Объединить блоки — записать сначала отсортированный блок *B1*, потом *B2, … , Bk*

Частичная сортировка лучше обычной в том случае, если в первом пункте у нас получится разбить последовательность на большое число блоков. Однако далеко не каждое такое разбиение подходит. Определите максимальное число блоков, на которое можно разбить исходную последовательность, чтобы сортировка отработала корректно.

Рассмотрим пример: *a = [3, 2, 0, 1, 4, 6, 5]*.  
Минимальный размер первого блока *B1* равен *4*. Если взять лишь первые два элемента, то отсортированная последовательность будет начинаться с двойки, что неправильно. Если взять первые три элемента, то последовательность будет начинаться с нуля, но после него сразу же пойдёт двойка. Первые четыре элемента уже гарантируют корректный префикс после объединения блоков. Четвёрку можно взять как самостоятельный блок из одного элемента. Последние два элемента надо объединить в третий блок. Таким образом:

*B1 = { 3, 2, 0, 1 }*  
*B2 = { 4 }*  
*B3 = { 6, 5 }*

В данном примере ответ равен *3*. Максимальное число блоков равно трём.

Формат ввода

В первой строке задано *n* — количество чисел для сортировки (*n ≤ 1000*).  
В следующей строке записаны числа от *0* до *n - 1*, которые надо разбить на блоки.

Формат вывода

Выведите максимальное количество блоков, на которое можно разбить данные при использовании метода Частичной сортировки.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  0 1 3 2 | 3 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 8  3 6 7 4 1 5 0 2 | 1 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5  1 0 2 3 4 | 4 |