

Обязательно ознакомьтесь с памяткой участника. Обратите внимание на следующие пункты:

1. Используйте **только** ввод с клавиатуры и вывод на экран. Никакие файлы `input.txt/output.txt` или другие использовать нельзя.
2. Не используйте в Паскале модуль `crt` (и, следовательно, функцию `clrscr`).
3. Программа **должна заканчивать** работу после вывода требуемых выходных данных. Не используйте дополнительный `read/readln` или `system("pause")` в конце программы.
4. Не выводите на экран какую-либо лишнюю информацию, например: `writeln("Vvedite n:").` Программа должна выводить на экран **только** то, что написано в секции **Формат выходных данных** условия задачи.
5. Не проверяйте входные данные на корректность. Жюри **гарантирует**, что они удовлетворяют всем требованиям из секции задачи **Формат входных данных** условия задачи.

Памятка участника олимпиады

Следуйте следующим правилам во время участия в олимпиаде:

1. Соревнование продлится **4 часа**.
2. Запрещается использовать любые электронные источники и носители информации, включая, но не ограничиваясь ими: портативные компьютеры, калькуляторы, радиотелефоны, пейджеры, дискеты, компакт-дискеты и прочее.
3. Запрещается общаться с другими участниками в любой форме.
4. Запрещается пользоваться Интернетом (за исключением доступа к тестирующей системе, см. пункт 13).
5. Решением является программа, написанная на одном из следующих языков программирования:
 - Pascal ABC — расширение `pas`
 - Free Pascal — расширение `pp`
 - C++ (Visual Studio 2010) — расширение `cpp`
 - C++14 (MinGW 6) — расширение `sxx`
 - Java 8 — расширение `java`
 - Python 3 — расширение `py`
 - C# (Visual Studio 2010, .NET 4.0) — расширение `cs`
6. На проверку отдается исходный текст программы, а не исполняемый файл. Программа должна полностью содержаться в одном файле, использование своих модулей и заголовочных файлов не допускается.
7. Программа должна читать входные данные только с клавиатуры и выводить результат на экран. Работать с файлами или какими-либо другими ресурсами компьютера запрещено.
8. Результаты работы программы проверяются автоматически, поэтому программа должна *точно соблюдать* формат вывода, указанный в условии. Не допускается вывод дополнительных сообщений. Ваша программа должна заканчивать свою работу после вывода ответа, а не «зависать» в ожидании ввода с клавиатуры.
9. Гарантируется, что входные данные будут соответствовать формату, указанному в условии. Все ограничения, заявленные в условии задачи, будут выполняться в тестах жюри. Проверять их в программе не имеет смысла. Ограничения по времени указаны для компьютера с тактовой частотой 3.2 ГГц.
10. Максимальный балл за каждую задачу указан в условии. Частичные решения будут оцениваться исходя из количества пройденных тестов.
11. Старайтесь использовать как минимум 32-х битные типы данных, если это позволяет ограничение по памяти (например, `longint` в Паскале).
12. Имя главного класса для программ на Java — `Solution` (см. пример программы на Java ниже).
13. Если вы участвуете через сайт <http://acm.sgu.ru/contest>, то вы самостоятельно отправляете решения в тестирующую систему. Логин и пароль для входа на сайт вы должны получить от преподавателя. Ознакомьтесь с инструкцией, размещенной на сайте.

14. В противном случае ваши решения будут собраны с вашего компьютера преподавателем по окончании работы. Для этого именуруйте файлы своих решений таким образом: `xxxxdd.ext`, где `xxxx` — это ваш четырехбуквенный идентификатор, `dd` — это номер задачи, а `ext` — это расширение вашего языка программирования. Например, имя файла для решения задачи 01 участника `abcd` на языке Pascal должно иметь вид `abcd01.pas`. Решения на Java следует именовать таким же образом, при компиляции вашей программы файл будет переименован в `Solution.java`.
15. Результаты будут доступны через 3 дня после окончания на сайте <http://acm.sgu.ru/olimp>, и также будут распространены через вашего учителя информатики.

Примеры решения задачи A+B

Ниже приведены решения задачи, в которой требуется считать два числа и вывести их сумму.

- Pascal:

```
var
  a, b: longint;
begin
  read(a, b);
  writeln(a + b);
end.
```

- C++:

```
#include <iostream>
using namespace std;

int main() {
  int a, b;
  cin >> a >> b;
  cout << a + b << endl;
  return 0;
}
```

- Java:

```
import java.util.Scanner;
import java.io.PrintWriter;

public class Solution {
  public static void main(String[] args) {
    Scanner scan
      = new Scanner(System.in);
    PrintWriter writer
      = new PrintWriter(System.out);
    int a = scan.nextInt();
    int b = scan.nextInt();
    writer.print(a + b);
    writer.close();
  }
}
```

- Python:

```
(a,b)=input().split()
print(int(a) + int(b))
```

- C#:

```
using System;
class Solution {
  static void Main() {
    String[] input
      = Console.ReadLine().Split(' ');
    Console.WriteLine(
      Convert.ToInt32(input[0]) +
      Convert.ToInt32(input[1])
    );
  }
}
```

Обратите внимание на следующую информацию

25-го сентября в 17:30 в 12-м корпусе СГУ (угол Вольской и Белоглинской) в 310 ауд. состоится организационное собрание городского кружка школьников по программированию. На собрание приглашаются школьники и их родители (можно только школьники).

Кружок будет работать в двух группах:

- младшая группа (8-11 класс): основы языка Паскаль, основные языковые конструкции, решение задач и упражнений, предполагается базовое знакомство с Паскаль в рамках школьного курса;
- старшая группа (8-11 класс): введение в алгоритмы и структуры данных, предполагается умение решать базовые задачи на Паскале/C++, зачисление в группу по результатам тестирования или олимпиад прошлого года.

На собрании будет подробно рассказано о структуре курсов, проведено тестирование школьников для деления на группы. Занятия будут вести сотрудники Центра олимпиадной подготовки, призеры этапов чемпионата мира по программированию.

Задача 1. Ресторан

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В новом ресторане морской кухни в Берляндии каждое блюдо стоит 750 бурлей. При этом в ресторане действует система скидок — за каждые 13 заказанных блюд клиент получает обратно 350 бурлей.

Николай пришел в ресторан и решил, что закажет n блюд. Определите количество бурлей, которые он потратит в ресторане, то есть определите величину $x - y$, где x — это суммарная стоимость всех заказанных блюд, а y — это количество бурлей, которые Николай получит обратно в соответствии с системой скидок.

Формат входных данных

В первой строке следует целое число n ($1 \leq n \leq 1000$) — количество блюд, которые Николай закажет в ресторане.

Формат выходных данных

Выведите целое число — количество бурлей, которые Николай потратит в ресторане.

Примеры

стандартный ввод	стандартный вывод
25	18400
4	3000
1000	723400

Замечание

В первом примере Николай получит только один возврат 350 бурлей за первые 13 блюд.

Во втором примере Николай не получит ни одного возврата 350 бурлей, так как он планирует заказать меньше 13 блюд.

Задача 2. Префиксы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У Николая есть строка s четной длины n , состоящая из латинских букв «a» и «b».

Николай хочет изменить свою строку таким образом, чтобы в любом её префиксе четной длины было поровну букв «a» и «b». Для этого Николай может производить следующие операции: выбрать букву в своей строке и заменить ее на другую, то есть заменить букву «a» на букву «b» или заменить букву «b» на букву «a». Использовать какие-либо другие буквы Николай не будет.

Под префиксом строки s понимается некоторая непустая последовательность подряд идущих первых букв строки s .

Определите минимальное количество операций, которое должен сделать Николай со строкой s , чтобы в любом префиксе четной длины строки s стало поровну букв «a» и «b».

Формат входных данных

В первой строке следует целое четное число n ($2 \leq n \leq 2 \cdot 10^5$) — длина строки s .

Во второй строке следует строка s длины n , состоящая из латинских букв «a» и «b».

Формат выходных данных

В первую строку выведите минимальное количество операций замен букв, которые Николай должен произвести, чтобы в

любом префиксе четной длины его строки стало поровну букв «a» и «b».

Во вторую строку выведите строку Николая после проведения им всех операций. Если ответов несколько, разрешается вывести любой из них.

Примеры

стандартный ввод	стандартный вывод
4 bbbb	2 abba
6 ababab	0 ababab
2 aa	1 ba
10 bbbabbbaab	2 abbaabbaab

Замечание

В первом примере Николай должен произвести две замены. Например, он может заменить первую букву «b» на букву «a» и заменить последнюю букву «b» на букву «a».

Во втором примере не нужно заменять никакие буквы, так как в любом префиксе четной длины исходной строки содержится поровну букв «a» и «b».

Задача 3. Стрельба по банкам

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вася недавно начал заниматься спортивной стрельбой из пистолета. Сегодня тренер предложил ему следующее задание. Он выставил на стол n банок в ряд. Банки пронумерованы слева направо, начиная с единицы. Чтобы закончить тренировку, Васе предстоит попасть в каждую из банок ровно по одному разу. При этом **порядок**, в котором Вася будет стрелять в банки, он выбирает **самостоятельно**.

Про i -ю банку Вася знает **параметр** a_i , а именно, если Вася уже попал в x банок до i -й и сейчас начнет стрелять в i -ю банку, то, чтобы попасть в эту банку, ему понадобится $(a_i \cdot x + 1)$ выстрел. Считайте, что если Вася начал стрелять в банку номер i , то пока он не попадет в нее, он не будет стрелять по другим банкам.

Определите такой порядок стрельбы по банкам, что Вася сделает минимально возможное количество выстрелов, чтобы попасть в каждую из n банок ровно по одному разу.

Формат входных данных

В первой строке следует целое число n ($2 \leq n \leq 1000$) — количество банок.

Во второй строке следует последовательность a_1, a_2, \dots, a_n ($1 \leq a_i \leq 1000$), где a_i равно параметру для i -й банки.

Формат выходных данных

В первую строку выведите минимальное количество выстрелов, которые должен сделать Вася, чтобы попасть в каждую из n банок ровно по одному разу.

Во вторую строку выведите последовательность из n **различных** целых чисел — номера банок в том порядке, в котором Вася должен по ним стрелять, чтобы минимизировать количество выстрелов. Если подходящих последовательностей несколько, разрешается вывести любую из них.

Примеры

стандартный ввод	стандартный вывод
3 20 10 20	43 1 3 2
4 10 10 10 10	64 2 1 4 3
6 5 4 5 4 4 5	69 6 1 3 5 2 4
2 1 4	3 2 1
3 3 4 3	12 2 1 3

Замечание

В первом примере Вася может сначала стрелять в первую банку. В нее он попадет с первого раза, так как до этого не попадал ни в одну банку. После этого он должен стрелять в третью банку. Чтобы попасть в нее он потратит $20 \cdot 1 + 1 = 21$ выстрел. После этого останется только вторая банка. Чтобы попасть в нее Вася сделает $10 \cdot 2 + 1 = 21$ выстрел. Таким образом, суммарно Вася сделает $1 + 21 + 21 = 43$ выстрела.

Во втором примере Васе не важно в какой последовательности стрелять по банкам, так как они все имеют одинаковое значение параметра.

Задача 4. Белый лист

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

На прямоугольном столе лежит белый лист бумаги. Он имеет прямоугольную форму, а его стороны параллельны сторонам стола. Если смотреть на стол сверху, и представить, что левая нижняя точка стола имеет координаты $(0, 0)$, а оси координат совпадают с левой и нижней сторонами стола, то левый нижний угол белого листа имеет координаты (x_1, y_1) , а правый верхний — (x_2, y_2) .

После этого на стол положили два черных листа бумаги. Стороны обоих черных листов параллельны сторонам стола. Координаты левого нижнего угла первого черного листа равны (x_3, y_3) , а правого верхнего — (x_4, y_4) . Координаты левого нижнего угла второго черного листа равны (x_5, y_5) , а правого верхнего — (x_6, y_6) .

Определите, осталось видно хотя бы часть белого листа, если смотреть на стол сверху, после того, как на стол положили два черных листа. Часть белого листа видно в том случае, если существует хотя бы одна точка, лежащая **строго внутри** белого листа и лежащая **строго вне** обоих черных листов.

Формат входных данных

В первой строке следуют четыре целых числа x_1, y_1, x_2, y_2 ($0 \leq x_1 < x_2 \leq 10^6, 0 \leq y_1 < y_2 \leq 10^6$) — координаты левого нижнего и правого верхнего углов белого листа.

Во второй строке следуют четыре целых числа x_3, y_3, x_4, y_4 ($0 \leq x_3 < x_4 \leq 10^6, 0 \leq y_3 < y_4 \leq 10^6$) — координаты левого нижнего и правого верхнего углов первого черного листа.

В третьей строке следуют четыре целых числа x_5, y_5, x_6, y_6 ($0 \leq x_5 < x_6 \leq 10^6, 0 \leq y_5 < y_6 \leq 10^6$) — координаты левого нижнего и правого верхнего углов второго черного листа.

Формат выходных данных

Если после того, как на стол положили два черных листа, осталось видно хотя бы часть белого листа, выведите «YES» (без кавычек). В противном случае, выведите «NO».

Примеры

стандартный ввод	стандартный вывод
2 2 4 4 1 1 3 5 3 1 5 5	NO
3 3 7 5 0 0 4 6 0 0 7 4	YES
5 2 10 5 3 1 7 6 8 1 11 7	YES
4 4 5 5 4 4 5 5 1 2 3 3	NO
10 10 11 11 10 10 11 11 10 10 11 11	NO
0 0 1000000 1000000 0 0 4999999 1000000 500000 0 1000000 1000000	YES

Замечание

В первом примере белый лист полностью будет покрыт черными листами.

Во втором примере часть белого листа будет видно, после того, как на стол положили два черных листа. Например, точка $(6.5, 4.5)$ лежит строго внутри белого листа и лежит строго вне обоих черных листов.

Задача 5. Мечи

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В подвале театра находилось n видов мечей, которые использовались во время представлений. Каждого вида мечей было **ровно** по x штук. В подвал проникли y человек, каждый из которых взял ровно по z мечей, причем каждый человек брал мечи только одного вида.

На следующий день директор театра обнаружил пропажу. Пересчитав мечи, директор понял, что в подвале осталось a_i мечей i -го вида.

Директор не помнит, сколько изначально мечей каждого вида было в подвале, а также не знает, сколько человек проникли в подвал и сколько мечей взял каждый из них. Вам предстоит определить **минимальное** количество людей y , которые могли проникнуть в подвал театра, а также по сколько мечей z взял каждый из них.

Формат входных данных

В первой строке следует целое число n ($2 \leq n \leq 2 \cdot 10^5$) — количество видов мечей.

Во второй строке следует последовательность a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^9$), где a_i равно количеству мечей i -го вида, которые остались в подвале после проникновения. Гарантируется, что найдется такая пара индексов (j, k) , что $a_j \neq a_k$.

Формат выходных данных

Выведите два целых числа y и z — минимальное количество людей, которые могли проникнуть в подвал, а также количество мечей, которое взял каждый из них.

Примеры

стандартный ввод	стандартный вывод
3 3 12 6	5 3
2 2 9	1 7
7 2 1000000000 4 6 8 4 2	2999999987 2
6 13 52 0 13 26 52	12 13

Замечание

В первом примере минимальное значение y равно 5, то есть минимальное количество людей, которые могли проникнуть в подвал, равно 5. Каждый из них взял по 3 меча, причем трое из них взяли по 3 меча первого вида, а двое из них взяли по 3 меча третьего вида.

Во втором примере минимальное значение y равно 7, то есть минимальное количество людей, которые могли проникнуть в подвал, равно 7. Каждый из них взял по 1 мечу первого вида.

Примеры

стандартный ввод	стандартный вывод
5 1 3 20 38 56	1 2 5 2 0
4 2132 506 999999999999999999 1000000000000000000	8 2 4 1
1 12345678	5
1 777	1

Замечание

Ответы на запросы из первого примера подробно разобраны в условии.

Задача 6. Числовая последовательность

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам задана *бесконечная* последовательность вида «112123123412345...», в которой следуют блоки всех последовательных целых чисел, записанных друг за другом. Первый блок содержит все числа от 1 до 1, второй — от 1 до 2, третий — от 1 до 3, ..., i -й блок содержит все числа от 1 до i .

Таким образом, первые 56 элементов последовательности равны «11212312341234512345612345671234567812345678912345678910». Элементы последовательности нумеруются с единицы. Например, 1-й элемент последовательности равен 1, 3-й элемент последовательности равен 2, 20-й элемент последовательности равен 5, 38-й элемент равен 2, 56-й элемент последовательности равен 0.

Перед вами стоит задача ответить на q независимых запросов. Для i -го запроса задано целое число k_i . Определите элемент последовательности, который находится в позиции k_i .

Формат входных данных

В первой строке следует целое число q ($1 \leq q \leq 500$) — количество запросов.

В i -й из следующих q строк следует целое число k_i ($1 \leq k_i \leq 10^{18}$) — описание очередного запроса.

Обратите внимание, что не менее чем в половине тестов $q = 1$ и $1 \leq k_i \leq 10^9$.

Формат выходных данных

Выведите q строк. В i -ю строку должна быть выведена цифра x_i ($0 \leq x_i \leq 9$) — ответ на запрос номер i , то есть x_i должно быть равно элементу последовательности, который находится в позиции k_i .

Задача 7. Wi-Fi

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вы живете в доме, состоящем из n комнат, расположенных в ряд. Комнаты пронумерованы слева направо, начиная с единицы.

Вы хотите подключить все n комнат к интернету. Каждая из комнат может быть подключена к интернету кабелем напрямую, стоимость подключения i -й комнаты равна i монет.

Про **некоторые** комнаты известно, что в них можно установить роутер. Стоимость установления роутера в i -й комнате равна i монет. При установке роутера в комнате i к интернету подключаются все комнаты с номерами от $(i - k)$ до $(i + k)$ включительно, где k — радиус действия роутера. Для всех роутеров значение k одинаково.

Определите минимальную стоимость подключения всех n комнат к интернету. Считайте, что количество комнат, в которые можно установить роутеры, не превосходит количества роутеров, которые у вас есть.

Формат входных данных

В первой строке следуют два целых числа n и k ($1 \leq n, k \leq 2 \cdot 10^5$) — количество комнат и радиус действия каждого роутера.

Во второй строке следует строка s длины n , состоящая только из нулей и единиц. Если i -й символ строки равен единице, то в i -й комнате можно установить роутер. Если i -й символ строки равен нулю, то в i -й комнате нельзя установить роутер.

Формат выходных данных

Выведите одно целое число — минимальную стоимость подключения всех n комнат к интернету.

Примеры

стандартный ввод	стандартный вывод
5 2 00100	3
6 1 000000	21
4 1 0011	4
12 6 000010000100	15

Замечание

В первом примере достаточно установить роутер в комнату 3, тогда все комнаты будут подключены к интернету. Стоимость подключения равна 3.

Во втором примере нельзя устанавливать роутеры ни в одну из комнат, поэтому все комнаты нужно подключать напрямую кабелем. Таким образом, стоимость подключения всех комнат равна $1 + 2 + 3 + 4 + 5 + 6 = 21$.

В третьем примере комнату 1 нужно подключить напрямую кабелем, а в комнату 3 установить роутер. Таким образом, стоимость подключения всех комнат равна $1 + 3 = 4$.

В четвертом примере нужно установить роутеры в комнаты 5 и 10. Тогда все комнаты будут подключены к интернету. Стоимость подключения равна $5 + 10 = 15$.