

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:
 - Free Pascal 2.6 — расширение `pas`
 - C++ (Visual Studio 2010) — расширение `cpp`
 - C++ (MinGW 4.7) — расширение `sxx`
 - Java 7 — 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 поздно вечером 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]));
  }
}
```

Задача 01. Поездка в такси (60 баллов)

Ограничение по времени: 0.5 секунды
Ограничение по памяти: 256 мегабайт

В этой задаче вам требуется написать программу, производящую расчет стоимости поездки в городском такси. Стоимость поездки зависит от времени ожидания пассажира и расстояния поездки. Пусть пассажира пришлось ждать t минут, и расстояние поездки оказалось равным s километров. Одна минута ожидания стоит x рублей, один километр пути — y рублей. Первые 10 минут ожидания пассажира бесплатны. Вам заданы t, s, x, y . Вычислите стоимость поездки.

Формат входных данных

В первой строке записаны четыре целых числа t, s, x и y , разделенные одиночными пробелами ($1 \leq t, s, x, y \leq 100$).

Формат выходных данных

Выведите одно число — стоимость поездки.

Примеры

ввод с клавиатуры	вывод на экран
15 6 1 20	125
8 10 2 15	150
20 5 3 10	80

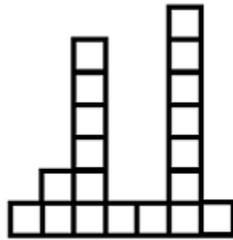
Примечание

В первом примере время ожидания пассажира составило 15 минут, из них 10 — бесплатны. За оставшиеся 5 минут пассажир должен заплатить по 1 рублю за минуту (получается 5 рублей). За 6 километров пути пассажир заплатит по 20 рублей за километр (то есть 120 рублей). Общая стоимость поездки составит 125 рублей. Во втором примере пассажир ничего не платит за ожидание, потому что его пришлось ждать всего 8 минут.

Задача 02. Забор (60 баллов)

Ограничение по времени: 0.5 секунды
Ограничение по памяти: 256 мегабайт

Перед домом Поликарпа находится забор, который состоит из n одинаковых по ширине досок, расположенных одна за другой слева направо. Высота i -ой доски составляет h_i метров, разные доски могут иметь различные высоты.



Забор для $n = 7$ и $h = [1, 2, 6, 1, 1, 7, 1]$

Поликарп приобрел рояль и теперь планирует способ занести рояль в свой дом. Для того, чтобы осуществить задуманное, ему придется выломать ровно три подряд идущих доски в заборе. Так как высокие доски выламывать сложнее, Поликарп хочет найти такие три последовательные доски, что сумма их высот минимальна.

Напишите программу, которая найдет номера трех последовательных досок с наименьшей суммой высот.

Формат входных данных

В первой строке входных данных содержится целое число n ($3 \leq n \leq 100$) — количество досок в заборе. Вторая строка содержит последовательность целых чисел h_1, h_2, \dots, h_n ($1 \leq h_i \leq 100$), где h_i — высота i -ой доски забора.

Формат выходных данных

Выведите такое j , что сумма высот досок j , $j + 1$ и $j + 2$ — наименьшая возможная. Если таких j несколько, то выведите любое из них.

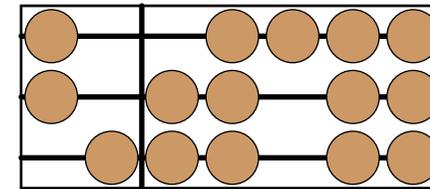
Примеры

ввод с клавиатуры	вывод на экран
7 1 2 6 1 1 7 1	3

Задача 03. Соробан (60 баллов)

Ограничение по времени: 0.5 секунды
Ограничение по памяти: 256 мегабайт

Как вы знаете, Япония является страной с едва ли не самым большим количеством электронных устройств на душу населения. С учетом этого кажется удивительным тот факт, что в начальной школе в Японии для обучения счету повсеместно используется деревянный *Соробан* — японский аналог счетов. Конечно, у этого явления есть свои веские причины, но мы не будем их обсуждать, а обратим свое внимание на устройство Соробана.



Соробан состоит из некоторого количества спиц, на которые нанизаны по пять костяшек. Будем считать, что спицы расположены горизонтально. На каждой спице одна костяшка (самая левая) отделена от остальных перегородкой. Эта костяшка называется *небесной*, а четыре остальных — *земными*. Каждая спица отвечает за представление одной цифры от 0 до 9. Значение цифры получается по следующему простому алгоритму:

- Положить значение цифры равным 0.
- Если небесная костяшка прижата вправо, то прибавить 5.
- Прибавить к результату количество земных костяшек, прижатых влево.

Таким образом, верхняя спица на рисунке представляет цифру 0, средняя — цифру 2, а нижняя — цифру 7. Будем считать, что верхняя спица означает младший разряд числа, поэтому число, представленное на рисунке — 720.

Напишите программу, которая выводит, как представляется на Соробане заданное вам двузначное число n .

Формат входных данных

В первой строке записано целое число n ($10 \leq n \leq 99$).

личных денег для аренды r велосипедов. Если школьники не могут взять напрокат ни одного велосипеда, то $r = s = 0$.

Примеры

ввод с клавиатуры	вывод на экран
2 2 10 5 5 7 6	2 3
4 5 2 8 1 1 2 6 3 7 5 2	3 8

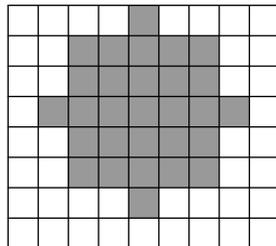
Примечание

В первом примере оба школьника могут взять напрокат велосипед. Например, они могут поделить общий бюджет пополам (по 5 рублей каждому). В этом случае одному из них придется доплатить 1 рубль из личных средств, а другому — 2 рубля. В сумме из личных средств будет потрачено 3 рубля. Такой вариант траты денег минимизирует расход личных средств.

Задача 06. Два круга (60 баллов)

Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Пусть задана таблица $n \times m$, заполненная целыми числами. Клетку в i -ой строке и j -ом столбце будем обозначать (i, j) . Таким образом, $(1, 1)$ является левой верхней клеткой таблицы, а (n, m) — правой нижней. Назовем кругом радиуса r с центром в клетке (i_0, j_0) множество таких клеток (i, j) , что $\sqrt{(i - i_0)^2 + (j - j_0)^2} \leq r$. Будем рассматривать только такие круги, которые не выходят за пределы таблицы, то есть у которых $r + 1 \leq i_0 \leq n - r$ и $r + 1 \leq j_0 \leq m - r$.



Круг радиуса 3 с центром в $(4, 5)$.

Найдите два таких непересекающихся круга заданного радиуса r , что сумма чисел в клетках, принадлежащим этим кругам, максимальна. Два круга пересекаются, если найдется клетка, принадлежащая обоим кругам. Поскольку может быть более одного способа выбрать пару кругов с максимальной суммой, то нас будет интересовать еще и количество таких пар. Посчитайте количество неупорядоченных пар кругов, то есть, к примеру, пара кругов радиуса 2 с центрами в $(3, 4)$ и $(7, 7)$ это та же самая пара, что и пара кругов радиуса 2 с центрами в $(7, 7)$ и $(3, 4)$.

Формат входных данных

В первой строке записано три целых числа n , m и r ($2 \leq n, m \leq 500$, $r \geq 0$). Каждая из следующих n строк содержит по m целых чисел в диапазоне от 1 до 1000 — элементы таблицы. Строки таблицы перечислены сверху вниз, элементы в строках — слева направо. Гарантируется, что существует хотя бы один круг радиуса r , не выходящий за пределы таблицы.

Формат выходных данных

Выведите два числа — максимальную сумму чисел в клетках, которые находятся в двух непересекающихся кругах, и количество пар непересекающихся кругов с максимальной суммой. Если не существует ни одной пары непересекающихся кругов, то выведите 0 0.

Примеры

ввод с клавиатуры	вывод на экран
2 2 0 1 2 2 4	6 2
5 6 1 4 2 1 3 2 6 2 3 2 4 7 2 5 2 2 1 1 3 1 4 3 3 6 4 5 1 4 2 3 2	34 3
3 3 1 1 2 3 4 5 6 7 8 9	0 0