

# Общая информация по задачам олимпиады

## Доступ к результатам проверки решений задач во время тура

В течение тура по каждой задаче можно отправить не более 40 решений и получить информацию о результатах оценивания решения на тестах жюри.

## Требования к программам

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ. В каждой задаче входные данные необходимо считывать из стандартного потока ввода, выходные данные необходимо выводить в стандартный поток вывода.

## Процесс тестирования

Перед решением задачи ознакомьтесь с системой оценки решения. Обратите внимание, в некоторых задачах очередная подзадача будет тестироваться, только если пройдены все тесты предыдущих подзадач.

## Сложность и порядок задач

Задачи муниципального этапа по информатике упорядочены примерно по возрастанию сложности. Полное решение каждой задачи оценивается в 100 баллов.

## Ограничения

| Задачи                        | Ограничение по времени | Ограничение по памяти | Получение результатов во время тура  |
|-------------------------------|------------------------|-----------------------|--|
| <b>А. Игральные кубики</b>    | 1 секунда              | 256 МБ                | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |
| <b>В. Лунная арифметика</b>   | 1 секунда              | 256 МБ                | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |
| <b>С. Рациональное дерево</b> | 1 секунда              | 256 МБ                | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |
| <b>Д. Охрана крепости</b>     | 1 секунда              | 256 МБ                | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |
| <b>Е. Миссия «Чак-чак»</b>    | 3.5 секунды            | 512 МБ                | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |

С результатами проверки решений задач, тестами, решениями жюри, а также письменным разбором задач можно ознакомиться после окончания тура на сайте <http://kpfu.ru/math/olimpiady-dlya-shkolnikov-i-studentov/olimpiady-shkolnikov-po-informatike>

## Задача А. Игральные кубики

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

*Закажите себе свой игральный кубик мечты!*

Традиционный игральный кубик — это кубик, на каждой из шести граней которого записаны числа от 1 до 6, по одному на каждой грани.

По желанию заказчика фабрика по производству настольных игр выпускает кубики с любыми числами на гранях. Кубик считается *правильным*, если суммы чисел на каждой двух противоположных гранях будут одинаковыми. Например, правильным будет классический кубик, у которого на гранях записаны числа 1, 2, 3, 4, 5, 6. Действительно, эти числа можно расположить на противоположных гранях требуемым образом:  $1 + 6 = 2 + 5 = 3 + 4$ .

Сможет ли фабрика выполнить заказ на выпуск правильных кубиков с заданными наборами чисел? На этот вопрос должна ответить ваша программа, которая для каждого набора из шести чисел определяет, возможно ли это.

### Формат входных данных

В первой строке задано целое число  $n$  ( $1 \leq n \leq 10^5$ ) — количество кубиков, выпускаемых фабрикой. В следующих  $n$  строках содержатся по шесть чисел  $a_1, a_2, a_3, a_4, a_5, a_6$  ( $1 \leq a_i \leq 10^2$ ), которые требуется записать на гранях соответствующего кубика.

### Формат выходных данных

Для каждой из  $n$  строк запишите *yes*, если из заданного набора чисел можно сформировать правильный кубик, и *no* — в противном случае.

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

| Подзадача | Баллы | Ограничения          | Необходимые подзадачи | Информация о проверке |
|-----------|-------|----------------------|-----------------------|-----------------------|
| 1         | 30    | $1 \leq n \leq 10$   |                       | баллы                 |
| 2         | 30    | $1 \leq n \leq 10^3$ | 1                     | баллы                 |
| 3         | 40    | $1 \leq n \leq 10^5$ | 1, 2                  | баллы                 |

### Пример

| стандартный ввод                | стандартный вывод |
|---------------------------------|-------------------|
| 2<br>7 5 6 8 6 7<br>7 6 3 4 1 5 | yes<br>no         |

### Замечание

В примере  $n = 2$ ; из чисел первого набора можно сформировать правильный кубик ( $8 + 5 = 7 + 6 = 7 + 6$ ), а из чисел второго — нельзя.

## Задача В. Лунная арифметика

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

Лунная арифметика была придумана в 2011 году несколькими математиками. Она представляет собой версию арифметики, в которой операции сложения и умножения над *цифрами* определяются как операции  $\max$  и  $\min$ . Другими словами, сумма двух цифр  $a$  и  $b$  равна наибольшей из них, а произведение чисел  $a$  и  $b$  равно наименьшей из них. Например, в лунной арифметике,  $2+7 = \max\{2, 7\} = 7$  и  $2 \times 7 = \min\{2, 7\} = 2$ .

Лунные арифметические операции над неотрицательными многозначными числами выполняются как в обычной арифметике, например:

$$\begin{array}{r} 169 \\ + 248 \\ \hline 269 \end{array} \qquad \begin{array}{r} 169 \\ \times 248 \\ \hline 168 \\ 144 \\ 122 \\ \hline 12468 \end{array} .$$

Вам необходимо для двух заданных чисел  $a$  и  $b$  вычислить их сумму и произведение в лунной арифметике.

### Формат входных данных

В первой строке записано целое число  $a$  ( $1 \leq a \leq 10^{100}$ ). Во второй строке записано целое число  $b$  ( $1 \leq b \leq 10^{100}$ ).

### Формат выходных данных

В строке запишите сумму и произведение чисел  $a$  и  $b$  в лунной арифметике.

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

| Подзадача | Баллы | Ограничения                 | Необходимые подзадачи | Информация о проверке |
|-----------|-------|-----------------------------|-----------------------|-----------------------|
| 1         | 20    | $1 \leq a, b \leq 9$        |                       | баллы                 |
| 2         | 40    | $1 \leq a, b \leq 99$       | 1                     | баллы                 |
| 3         | 40    | $1 \leq a, b \leq 10^{100}$ | 1, 2                  | баллы                 |

### Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 169<br>248       | 269 12468         |
| 54<br>321        | 354 3321          |



## Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 3 4              | RLRR              |
| 4 3              | RRLR              |
| 8 6              | RRLR              |

## Задача D. Охрана крепости

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

На территории древней крепости расположены  $n$  складов, между которыми проложено  $m$  двусторонних дорог, длина каждой дороги 1 километр. Каждый склад имеет уникальный номер от 1 до  $n$ , а дороги, соединяющие некоторые из складов, также имеют свои номера от 1 до  $m$ . Например, дорога с номером  $i$  соединяет склады с номерами  $a_i$  и  $b_i$ .

На некоторых из этих складов размещены охранники, которых всего  $k$ . Каждый охранник имеет свой уникальный номер от 1 до  $k$ , располагается на определённом складе  $p_i$  и обладает запасом выносливости  $h_i$ . Запас выносливости охранника указывает на максимальное расстояние в километрах, которое он может преодолеть, чтобы охранять склады.

Склад с номером  $v$  считается охраняемым, если хотя бы один охранник может добраться до него, не превышая своей выносливости. Это значит, что расстояние между складом  $v$  и складом  $p_i$  охранника не должно превышать  $h_i$ . Расстояние между складами определяется как минимальное количество дорог, необходимых для достижения одного склада из другого.

Ваша задача — найти все охраняемые склады.

### Формат входных данных

Первая строка содержит три целых числа  $n$ ,  $m$  и  $k$  — количество складов, количество дорог и количество охранников соответственно ( $1 \leq n, m, k \leq 2 \cdot 10^5$ ).

Следующие  $m$  строк содержат описание дорог, соединяющих склады. Каждая из  $m$  строк содержит два целых числа  $a_i$  и  $b_i$  ( $1 \leq a_i, b_i \leq n$ ) — номера складов, связанных дорогой  $i$  длиной в 1 километр.

Следующие  $k$  строк описывают выносливость охранников. Каждая строка содержит два целых числа  $p_i$  и  $h_i$  ( $1 \leq p_i \leq n, 1 \leq h_i \leq 2 \cdot 10^5$ ), где  $p_i$  — номер склада, где располагается охранник, и  $h_i$  — его выносливость. Все значения  $p_i$  различны.

### Формат выходных данных

В первой строке выведите количество охраняемых складов.

Во второй строке запишите в порядке возрастания номера всех охраняемых складов.

Если ни один склад не охраняется, выведите пустую строку.

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены. Обозначим через  $G$  граф, у которого вершинами являются склады, а рёбрами — дороги, соединяющие некоторые из этих складов.

| Подзадача | Баллы | Ограничения   | Необходимые подзадачи | Информация о проверке |
|-----------|-------|---|-----------------------|-----------------------|
| 1         | 10    | $1 \leq n \leq 2000$ ,<br>граф $G$ — бамбук         |                       | баллы                 |
| 2         | 15    | $1 \leq n \leq 2000$ ,<br>граф $G$ — дерево         | 1                     | баллы                 |
| 3         | 15    | $1 \leq n \leq 2 \cdot 10^5$ ,<br>граф $G$ — бамбук | 1                     | баллы                 |
| 4         | 25    | $1 \leq n \leq 2 \cdot 10^5$ ,<br>граф $G$ — дерево | 1, 2, 3               | баллы                 |
| 5         | 15    | $1 \leq n, m \leq 5000$                             | 1, 2                  | баллы                 |
| 6         | 20    | $1 \leq n, m \leq 2 \cdot 10^5$                     | 1, 2, 3, 4, 5         | баллы                 |

Бамбуком называется дерево, степень вершин в котором не превосходит 2.

## Примеры

| стандартный ввод   | стандартный вывод |
|--|-------------------|
| 5 5 2<br>1 2<br>2 3<br>2 4<br>3 5<br>1 5<br>1 2<br>5 1                                       | 5<br>1 2 3 4 5    |
| 3 0 1<br>1 3   | 1<br>1            |
| 10 10 2<br>1 2<br>5 1<br>1 6<br>2 4<br>5 2<br>10 2<br>8 5<br>8 6<br>9 6<br>7 9<br>3 2<br>8 1 | 4<br>3 5 6 8      |

## Задача Е. Миссия «Чак-чак»

|                         |                   |
|-------------------------|-------------------|
| Имя входного файла:     | стандартный ввод  |
| Имя выходного файла:    | стандартный вывод |
| Ограничение по времени: | 3.5 секунд        |
| Ограничение по памяти:  | 256 мегабайт      |

На улицах IT-города Иннополиса республики Татарстан можно легко встретить робота-доставщика. Такой робот развозит на заказ любимый всеми чак-чак по  $n$  адресам жителей города. Маршрут его движения всегда состоит из нескольких прямолинейных отрезков, соединённых под прямым углом, что, по задумке разработчиков, обеспечивает безопасность как для горожан, так и для самого робота. Робот может обойти все  $n$  пунктов заказа в любом порядке, начав движение из произвольной точки города.

Ваша задача — составить маршрут минимальной длины, позволяющий роботу посетить все  $n$  адресов заказа.

### Формат входных данных

Первая строка содержит целое  $n$  ( $2 \leq n \leq 12$ ) — количество пунктов заказа.

В каждой из  $n$  следующих строк указаны два целых числа  $x_i$  и  $y_i$  — координаты соответствующих адресов ( $0 \leq x_i, y_i \leq 10^6$ ). Гарантируется, что все пункты заказа находятся в различных точках, то есть  $(x_i, y_i) \neq (x_j, y_j)$  для любых  $i \neq j$  ( $1 \leq i, j \leq n$ ).

### Формат выходных данных

Выведите длину кратчайшего маршрута, который позволит роботу посетить все адреса в любом порядке, с абсолютной или относительной погрешностью не более  $10^{-6}$ .

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

| Подзадача | Баллы | Ограничения | Необходимые подзадачи | Информация о проверке |
|-----------|-------|-------------|-----------------------|-----------------------|
| 1         | 10    | $n = 2$     |                       | баллы                 |
| 2         | 15    | $n \leq 3$  | 1                     | баллы                 |
| 3         | 25    | $n \leq 8$  | 1, 2                  | баллы                 |
| 4         | 15    | $n \leq 10$ | 1, 2, 3               | баллы                 |
| 5         | 35    | $n \leq 12$ | 1, 2, 3, 4            | баллы                 |

### Примеры

| стандартный ввод              | стандартный вывод |
|-------------------------------|-------------------|
| 3<br>1 2<br>0 1<br>3 0        | 4.242640687119    |
| 4<br>5 3<br>1 4<br>2 6<br>6 0 | 11.156638751671   |



## Замечание

На рисунке 1 указан один из кратчайших маршрутов обхода четырёх адресов из примера 2, его длина приблизительно равна 11.156638751671.

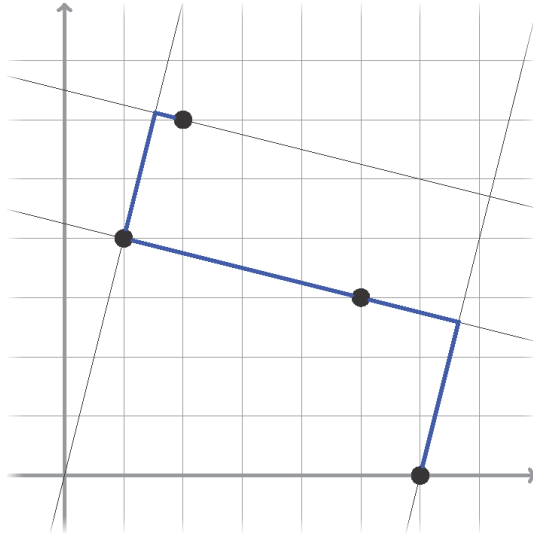


Рис. 1: Пример кратчайшего маршрута обхода всех пунктов заказа.