

## Задача 1. Тестовые данные. 40 баллов

**ВАЖНО!** В этой задаче не нужно писать программу.

Вам дана программа для подсчёта количества и суммы чётных (even) и нечётных (odd) чисел, вводимых пользователем, при этом вводится не более 10 целых чисел.

### Входные данные для программы

Несколько целых чисел могут разделяться произвольным количеством пробелов, табуляций, могут располагаться на нескольких строках. Каждое число может быть от -999 до 999. Количество чисел не более 10.

### Выходные данные для программы

Если пользователь ввёл больше 10 чисел, либо если любое из введённых чисел меньше -999 либо больше 999, либо если пользователь ввёл **не целое число**, программа должна вывести:

```
ERROR
```

Как только ввод данных закончится, программа должна вывести в стандартный вывод статистику сперва по чётным (even), а потом по нечётным числам (odd):

```
Sum      of      <N>      even      numbers      is      <X>
Sum of <M> odd numbers is <Y>
```

Вместо <M>, <N>, <X>, <Y> программа должна вывести соответствующее число.

Если пользователь не ввёл ни одного числа, программа должна вывести:

```
NO NUMBERS
```

Если пользователь не ввёл ни одного чётного, либо нечётного числа, то программа **не должна выводить** соответствующую строку, вроде:

```
Sum of 0 even numbers is 0.
```

**Изучите как работает данная вам правильная программа.**

### В задаче требуется:

Предоставить набор файлов с именами `input<N>.txt` и `output<N>.txt`, где <N> - некоторое натуральное число. Файл `input<N>.txt` содержит входные данные, а `output<N>.txt` - соответствующие выходные данные программы.

В тестировании будут участвовать несколько версий программ, часть из которых правильные, а часть - неправильные.

**Правильной** считается программа, полностью соответствующая требованиям задания.

**Неправильной** считается программа, которая не соответствует всем, либо некоторым требованиям задания.

Набор подготовленных вами файлов должен быть таким, чтобы:

- **все правильные** программы для каждого `input<N>.txt` вывели в стандартный вывод **результат, совпадающий** с `output<N>.txt`;
- **каждая из неправильных** программ **хотя бы для одного** из входных `input<N>.txt` выдала **результат, не совпадающий** с `output<N>.txt`.

## Задача 2. Робот-художник и его орнамент. 60 баллов

В одной крутой школе директор-математик решил украсить стену холла авангардным орнаментом. Для этого используется робот-художник, который умеет наносить орнамент в форме прямоугольного треугольника по заданным координатам.

Директор дал задание ученикам: определить координаты точек для нанесения орнамента. Ученики нарисовали план стены и отметили там точки. Помогите ученикам определить, все ли они сделали правильно?

Необходимо написать программу, которая проверит, можно ли из трех точек на плоскости построить прямоугольный треугольник.

### Входные данные

Программа получает на вход координаты трех точек на плоскости. Координаты точек подаются каждая с новой строки. Все координаты являются целыми числами от -100 до 100 включительно.

### Выходные данные

Программа должна выводить ответ "YES", если прямоугольный треугольник построить можно и "NO", если нельзя.

**Предусмотреть ситуацию**, когда входные данные некорректны: координаты больше 100 или меньше -100, не являются целыми числами. В этом случае программа должна выдавать сообщение об ошибке "ERROR".

### Справочная информация

1) Если даны две точки на плоскости  $A(x_1; y_1)$  и  $B(x_2; y_2)$ , то квадрат длины отрезка  $AB$  вычисляется по формуле:  $AB^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$

2) Для прямоугольного треугольника со сторонами длиной  $a$ ,  $b$  и  $c$  выполняется равенство  $a^2 + b^2 = c^2$ , при этом  $a < c$ ,  $b < c$ .

### Примеры возможных входных и выходных данных

Ввод	Вывод	Комментарий
1 1 5 1 1 4	YES	Даны три точки $A(1;1)$ , $B(5;1)$ и $C(1;4)$ , тогда $a^2 = AB^2 = (1-5)^2 + (1-1)^2 = 16$ $b^2 = AC^2 = (1-1)^2 + (1-4)^2 = 9$ $c^2 = BC^2 = (5-1)^2 + (1-4)^2 = 25$ действительно, $c^2 = a^2 + b^2$
1 1 3 1 2 3	NO	равенство $c^2 = a^2 + b^2$ не получается ни при каких сочетаниях точек

Ввод	Вывод	Комментарий
1 -500 1 -1 2 1	ERROR	У точки $A(1;-500)$ вторая координата меньше, чем -100
1 5 F 6 2 3	ERROR	'F' не является числом

### Задача 3. Эстафета. 80 баллов

В стране Берляндии наступил Новый Год. В честь праздника две лучших в стране команды биатлонистов, “Подснежники” и “Васильки”, решили соревноваться друг с другом и устроили эстафету. К сожалению, незадолго до праздника команду “Васильки” покинул тренер и двое лучших биатлонистов, и на эстафете они потерпели полный разгром.

Журналист Петя случайно узнал об этом и решил опубликовать сенсационную новость в первой газете страны: “Команда Подснежники побила рекорд, опередив соперников на финише на  $N$  метров!”. Однако, перед выпуском новости Пете нужно подсчитать, на сколько именно метров команда “Васильки” отставала от команды “Подснежники” в момент, когда “Подснежники” финишировали.

К счастью, Петя тщательно записал все участки дистанции и время, за которое команды проходили эти участки во время эстафеты. Также он заметил, что каждый спортсмен проходил свой участок с постоянной скоростью. Теперь Пете нужна ваша помощь: напишите для него программу, определяющую, на сколько метров лучшая из двух команд опередила худшую. Торопитесь, Пете нужно опубликовать новость уже сегодня!

Команды вышли со старта в одно и то же время. Когда первая команда финишировала, спортсмен из второй команды мог находиться где-то между началом и концом своего участка.

#### Входные данные

Программа получает на вход два числа  $N$  и  $T$ , где  $N$  - количество участков дистанции  $0 < N \leq 6$ , а  $T$  - время прохождения эстафеты первой командой (положительное целое число,  $0 < T \leq 1000$ ). Далее следует  $N$  пар положительных целых чисел, каждая пара содержит длину участка (в метрах) и время второй команды (в секундах).

Каждое число вводится на отдельной строке, а значения чисел могут быть от 1 до 1000.

**Совет:** вы можете обрабатывать данные по мере их поступления, либо использовать массивы.

#### Выходные данные

Программа должна вывести отставание второй команды в метрах с округлением в большую сторону (журналист Петя любит преувеличивать).

**В случае ошибки во входных данных** программа выводит ERROR. Это происходит, если указанное число участков превышает 6, либо вместо ожидаемого положительного числа введено не числовое значение, либо какое-то из чисел превышает верхний предел 1000, является нулем или отрицательным, либо оказывается, что вторая команда пришла к финишу раньше первой.

### Справочная информация

Прохождение спортсменами их участков в момент финиша первой команды могло выглядеть так (серым цветом отмечена пройденная дистанция):

№ участка	1	2	3	4	5
“Подснежники”					
“Васильки”					

Длина участка равна скорости, умноженной на время:  $s = v \cdot t$ .

Из этого следует, что скорость движения спортсмена равна длине его участка, поделённой на время:  $v = s / t$ .

Если спортсмен успел пройти только часть участка, то, зная затраченное на участке время  $t_1$ , мы можем вычислить пройденное расстояние:  $s_1 = v \cdot t_1 = (s / t) \cdot t_1$ . При расчёте в целых числах деление на  $t$  произойдёт с округлением в меньшую сторону.

Оставшуюся длину участка можно определить как  $s_2 = s - s_1 = s - (s \cdot t_1) / t$ . Поскольку пройденная часть участка будет округлена в меньшую сторону, после вычитания оставшаяся длина участка будет округлена в большую сторону.

### Примеры возможных входных и выходных данных

Ввод	Вывод	Комментарий
1 5 50 10	25	Вторая команда отстала на 5 секунд на 1 участке, скорость на участке составила 5 м/с, итого отставание 25 метров
2 5 20 4 31 6	26	Вторая команда отстала на 5 секунд, в момент финиша первой проходила 2-й участок (прошла 1 секунду, т.е. с округлением 5 метров), и 26 метров осталось пройти до финиша.
0 0	ERROR	Введено недопустимое число участков дистанции
1 100 1001 2000	ERROR	Два числа превышают 1000
1 5 50 2	ERROR	Вторая команда пришла раньше первой