

# Муниципальный этап олимпиады по информатике (7-8 класс)

Во всех задачах полностью правильное решение, укладывающееся в ограничения по времени и памяти, получает 100 баллов.

## 1. Илья Муромец

Ограничения: время – 200мс, память - 32МБ

Вышел Илья Муромец из леса на дорогу, ведущую из Киева в Чернигов, и увидел камень, на котором было написано «До села Калиновки - 120 верст, до реки Смородины - 50 верст, до Киева...». Остальная часть надписи на камне была неразборчива, но Илья вспомнил расстояние от Киева до села и до реки и легко определил, что от камня с надписью до Киева 200 верст.

В первой строке содержатся четыре целых числа, разделенные пробелами — расстояние от села Калиновки до Киева  $A$ , расстояние от реки Смородины до Киева  $B$  ( $1 \leq A < B \leq 1000$ ), расстояние от камня до села Калиновки  $X$  и расстояние от камня до реки Смородины  $Y$  ( $1 \leq X, Y \leq 1000$ ). Гарантируется, что числа  $A, B, X, Y$  соответствуют какому-то возможному расположению камня на дороге.

Вывести одно целое число — расстояние от камня с надписью до Киева.

Пример ввода	Пример вывода
80 150 120 50	200

## 2. Алгоритм

Ограничения: время – 200мс, память - 32МБ

Реализуйте на одном из языков программирования алгоритм, представленный на схеме.

В первой строке ввода содержатся два целых числа  $A$  и  $B$  ( $1 \leq A \leq B \leq 10^9$ ).

Вывести одно целое число – значение  $K$  после завершения работы алгоритма.

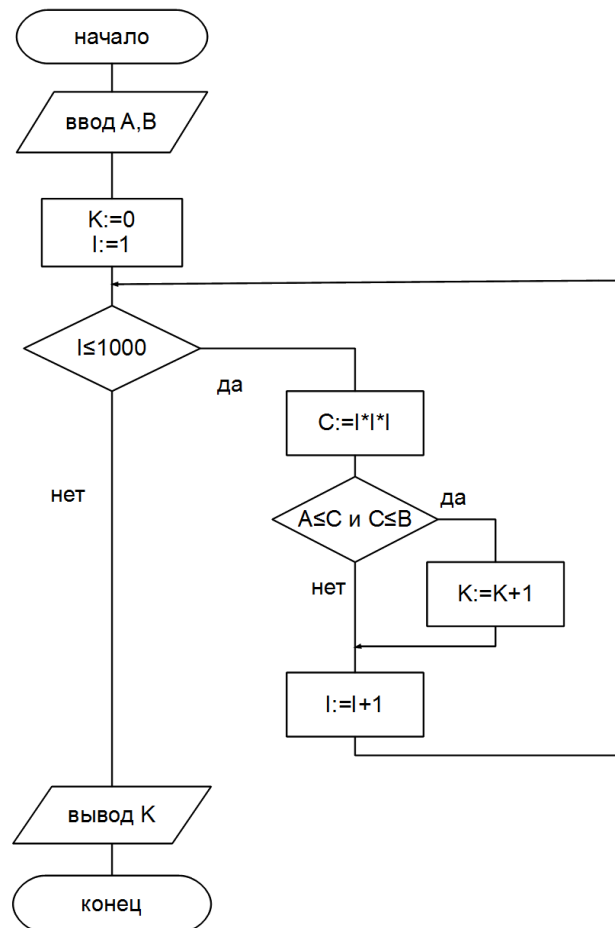
Пример ввода	Пример вывода
10 100	2

## 3. Автомобиль-робот

К 2020 году многие автокомпании планируют создать автомобили, которые могут самостоятельно доставить пассажиров из точки  $A$  в точку  $B$ .

Чтобы понять трудности, стоящие перед разработчиками программного обеспечения для автомобиля-робота, напишите программу для управления роботом в следующей задаче. Роботу

нужно проехать по двухполосной дороге с односторонним движением длиной 10 единиц, на которой припаркованы несколько машин (см. рис.). Машины могут стоять на любой полосе, но путь по дороге всегда существует. Первоначально робот находится в начале дороги на левой полосе. Робот может выполнять следующие команды:  $\wedge$  – ехать вперед,  $<$  – сдвинуться в полосу, расположенную левее текущей полосы движения,  $>$  – сдвинуться в полосу, расположенную правее. После команды можно указать число – количество повторений этой команды. Например,  $\wedge 5$  – ехать



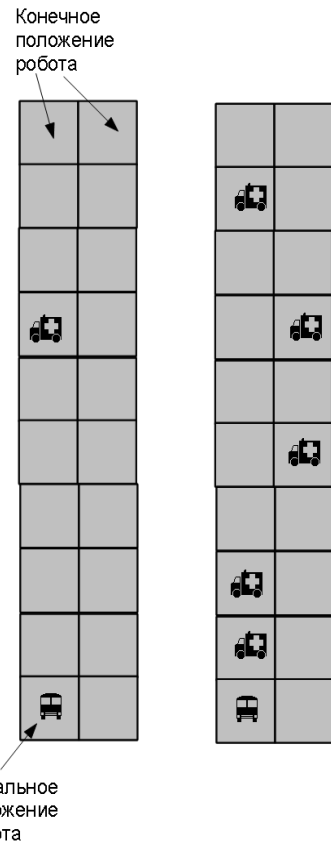
вперед 5 единиц. Несколько команд можно соединить в группу с помощью круглых скобок, тогда число после закрывающейся скобки будет показывать количество повторений этой группы, например,  $(\wedge^{\wedge} <) 3$ . Для проверки возможности движения в каком-то направлении используется команда  $?$ , после которой указывается команда движения  $\wedge$ ,  $<$ ,  $>$ . Если движение в этом направлении возможно, то команда выполняется. Иначе в программе ищется символ  $|$  (вертикальная черта) и выполняется команда после него. Например,  $?^{\wedge}|<$  – пытаемся ехать вперед, если нельзя, то сдвигаемся влево. Можно соединить несколько проверок в последовательность, например,  $?^{\wedge}|?<|>$ .

Робот должен из начального положения проехать в любую клетку на другой стороне дороги и там остановиться. При этом нельзя врезаться в припаркованные машины и выезжать за пределы дороги. Например, для левого рисунка решением является следующая программа:  $\wedge 5 > \wedge 4$ .

25 баллов вы можете получить за программу, которая позволит проехать роботу по дороге на правом рисунке.

100 баллов вы можете получить за универсальную программу, которая позволит роботу проехать по дороге с любой расстановкой препятствий (гарантируется, что путь существует).

В качестве решения вы должны выслать одну строку, содержащую программу для робота, при этом в качестве языка программирования вы должны указать Free Pascal.



## 4. Забор

Ограничения: время – 200мс, память - 32МБ

Петя решил сделать небольшой забор из нескольких имеющихся у него досок разной длины. Из эстетических соображений разница в длине планок забора не должна превышать некоторой величины  $D$ . Петя может разрезать доску на любое количество планок, имеющих любую длину. Необязательно, чтобы их длина была целым числом или планки из одной доски имели одинаковую длину. Но после разрезания нельзя выбрасывать ничего - все отрезанные от доски части становятся планками забора. Так как Пете приходится пилить доски ножовкой (вручную), то он хочет минимизировать количество распилов.

Напишите программу, которая подсчитает минимальное количество распилов для заданного набора досок на планки, при котором разница в длине между самой длинной и самой короткой планкой не будет превышать  $D$ .

Первая строка ввода содержит два целых числа — количество досок  $N$  ( $2 \leq N \leq 10$ ) и ограничение по разнице длин планок  $D$  ( $100 \leq D \leq 1000$ ). Во второй строке содержится  $N$  целых чисел в диапазоне от 1000 до 100000 — длины досок.

Вывести одно целое число — минимальное количество распилов.

Пример ввода	Пример вывода
2 100 1600 1100	3

Пояснение к примеру: Петя может первую доску распилить на планки длиной 500, 500 и 600 с помощью двух распилов, а вторую доску - на две планки длиной 500 и 600. Это не единственный вариант. Можно распилить вторую доску на части длиной 501 и 599 или 502.36 и 597.64.

Характеристика тестов к задаче: в 40% тестов  $N=2$ , в 60% тестов  $N=10$ .