

Муниципальный этап олимпиады по информатике (7-8 класс)

24 ноября 2022

1. Кузнечик

Кузнечик находится на числовой прямой в точке 0 и может совершать два вида прыжков — большой (длиной K) и маленький (длиной 1). Из точки с координатой x кузнечик может попасть в точки с координатой $x+K$ и $x-K$, сделав большой прыжок, и в точки с координатой $x+1$ и $x-1$, сделав маленький прыжок.

Определите минимальное количество прыжков, которое необходимо сделать кузнечику, чтобы попасть в точку с координатой X из точки с координатой 0.

Ввод содержит два целых числа X, K ($-10^9 \leq X \leq 10^9, 2 \leq K \leq 10^9$), по одному числу в строке — координата точки, куда нужно попасть кузнечику, и длина большого прыжка.

Вывести одно целое число — минимальное количество прыжков..

Пример ввода 1	Пример вывода 1
2 5	2
Пример ввода 2	Пример вывода 2
14 5	4

Пояснение к примеру 1: кузнечик делает 2 маленьких прыжка направо.

Пояснение к примеру 2: сначала кузнечик делает 3 больших прыжка и попадает в точку с координатой 15, затем 1 маленький прыжок налево и попадает в точку с координатой 14.

Система оценки и описание подзадач

Подзадача 1 (60 баллов)

$1 \leq X \leq 1000, 2 \leq K \leq 1000$

В этой подзадаче 6 тестов, каждый тест оценивается в 10 баллов. Баллы за каждый тест начисляются независимо..

Подзадача 2 (24 баллов)

$-1000 \leq X \leq 0, 2 \leq K \leq 1000$

Необходимые подзадачи: 1.

В этой подзадаче 3 теста, каждый тест оценивается в 8 баллов. Баллы за каждый тест начисляются независимо..

Подзадача 3 (16 баллов)

$-10^9 \leq X \leq 10^9, 2 \leq K \leq 10^9$

Необходимые подзадачи: 1, 2.

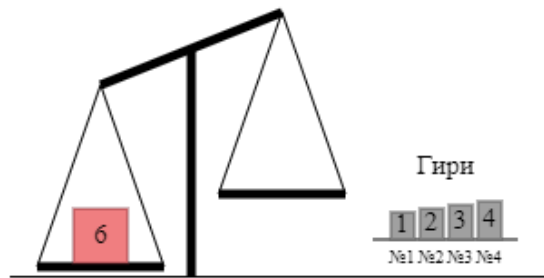
В этой подзадаче 4 теста, каждый тест оценивается в 4 балла. Баллы за каждый тест начисляются независимо.

По запросу сообщается результат окончательной проверки на каждом тесте.

2. Весовщик

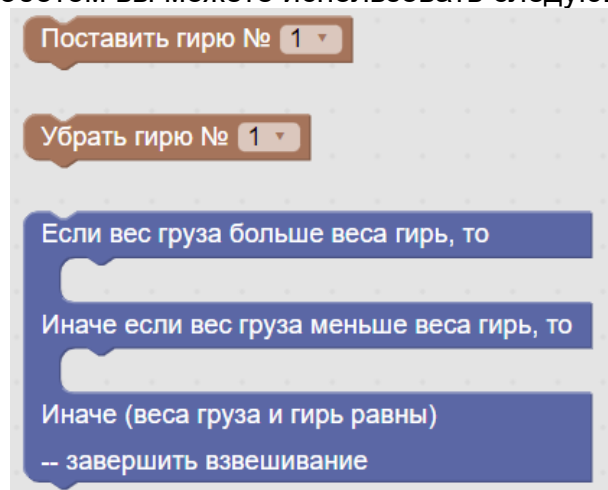
Напишите в Blockly программу для робота, который выполняет взвешивание груза весом от 1 до 10, выставя гири на чашечных весах.

Выполнено действий: 0



После выполнения программы груз на левой чашке должен быть уравновешен набором гирь на правой чашке.

Для управления роботом вы можете использовать следующие команды:



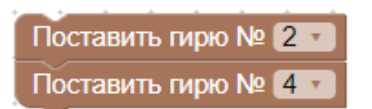
Первые две команды ставят гирю с указанным номером на чашку весов или убирают её.

Если выполнение команды невозможно, то она игнорируется.

Третья команда позволяет выбрать дальнейшие шаги в зависимости от положения чашек весов или завершить взвешивание, если вес груза и суммарный вес гирь совпадают.

Первые две команды будем называть действиями, так как при их выполнении происходит перемещение гирь на чашку весов или на стол, в отличие от третьей команды, которая делает выбор, но не требует физических действий.

Например, чтобы уравновесить груз с весом 6, нужно выполнить следующие команды:



Для запуска программы используйте кнопки:

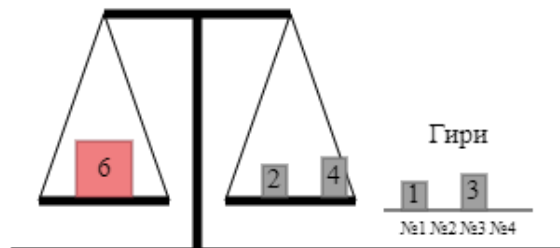


Первая кнопка позволяет отправить ваше решение для проверки в проверяющую систему соревнований, вторая кнопка выполняет запуск программы локально, третья — пошаговое выполнение или временная остановка программы, четвертая — завершение выполнения программы, после которой программа будет выполняться сначала.

Например, после запуска на выполнение программы, показанной выше, состояние весов изменится на

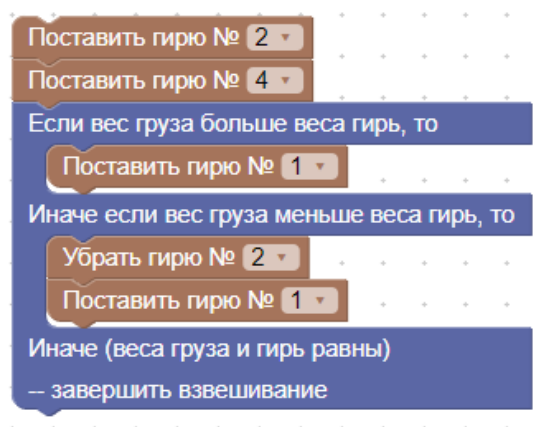
Вес груза:

Выполнено действий: 2

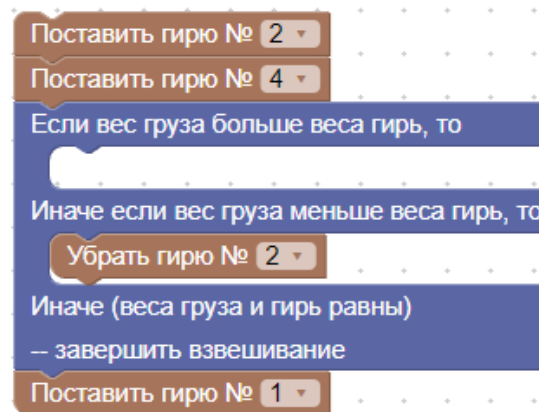


В поле "вес груза" можно задать другой вес груза от 1 до 10.

Например, программа, определяющая вес груза из набора 5,6,7 может выглядеть так:



Последняя команда "Поставить гирю №1" в команде выбора совпадает, поэтому её можно переставить после команды выбора.



Это не уменьшит количество выполняемых действий для уравнивания груза, только количество команд в программе.

В 4-й подзадаче вы можете попытаться написать программу, которая выполняет не более 5 действий для уравнивания любого груза весом от 1 до 10.

Система оценки и описание подзадач

Подзадача 1 (30 баллов)

Вес груза 1, 2 и 3.

В этой подзадаче 3 теста, каждый тест оценивается в 10 баллов. Баллы за каждый тест начисляются независимо.

Подзадача 2 (15 баллов)

Вес груза 5, 6 и 7.

Необходимые подзадачи: 1.

В этой подзадаче 3 теста, каждый тест оценивается в 5 баллов. Баллы за каждый тест начисляются независимо.

Подзадача 3 (40 баллов)

Вес груза 4, 8, 9 и 10.

Необходимые подзадачи: 1,2.

В этой подзадаче 4 теста, каждый тест оценивается в 10 баллов. Баллы за каждый тест начисляются независимо.

Подзадача 4 (15 баллов)

В этой подзадаче 1 тест. Программа запускается 10 раз для всех весов груза от 1 до 10.

Необходимые подзадачи: 1,2,3.

Количество действий для уравнивания груза не должно превышать 5.

По запросу сообщается результат окончательной проверки на каждом тесте.

Для написания программы для робота используется специальная версия среды Blockly. Вы можете использовать полную среду Blockly при решении других задач этого соревнования, выбрав пункт Blockly в информационном меню задачи.

3. Алгоритм

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

Первая строка ввода содержит одно целое число N , вторая строка ввода содержит одно целое число K ($1 \leq K \leq N \leq 1000000000$).

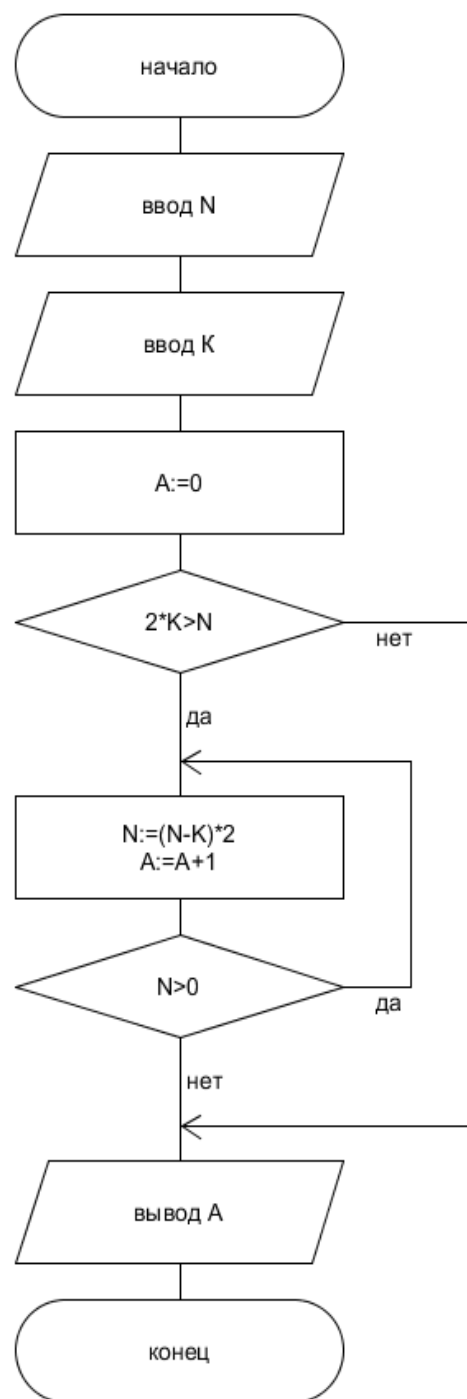
Вывести одно целое число — вычисленный ответ.

Пример ввода 1	Пример вывода 1
20 11	4
Пример ввода 2	Пример вывода 2
20 9	0

Система оценки

В этой задаче 5 тестов, каждый тест оценивается в 20 баллов. Баллы за каждый тест начисляются независимо.

По запросу сообщается результат окончательной проверки на каждом тесте.

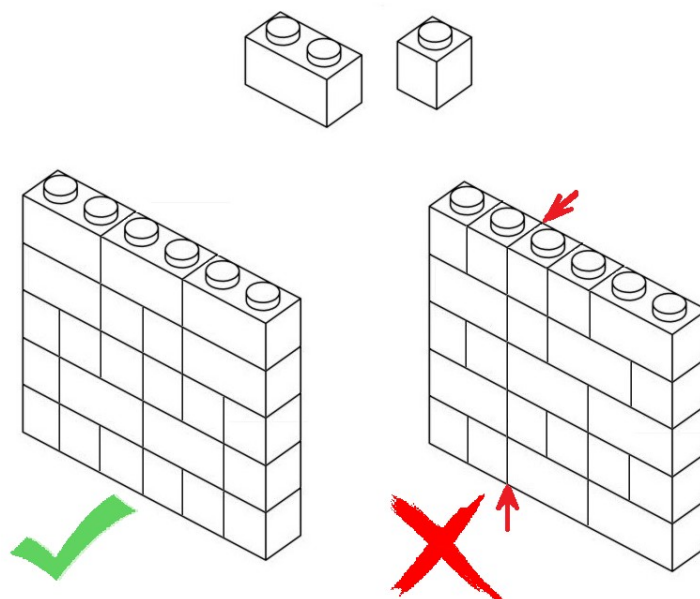


4. Идеальная стена

Саша хочет собрать стену из деталей Лего. Все детали имеют одинаковую высоту и делятся на 2 вида: "квадратные" размера 1 на 1 и "прямоугольные" размера 1 на 2 (изображены на рисунке). Каждая деталь надежно прикрепляется к соседям верхней и нижней гранью, но никак не скрепляется с боковыми соседями.

Чтобы стена была идеальной, должны выполняться следующие условия:

1. Все детали должны быть использованы. Нельзя оставлять детали не прикрепленными к стене.
2. Стена должна иметь вид прямоугольника толщины 1 без отверстий и выступов.
3. Стена не должна разваливаться на части. То есть, любой вертикальный разрез в пределах стены должен пересекаться с какой-нибудь деталью, которая не даёт стене распадаться по линии этого разреза. Примеры подходящей и неподходящей стен приведены на рисунке.



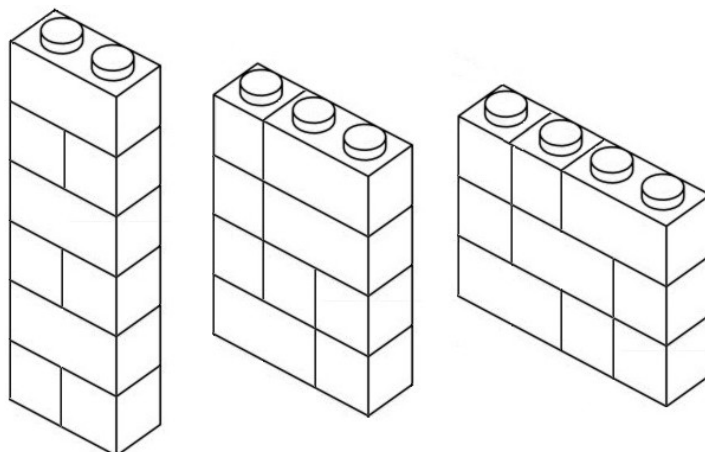
Для заданного количества "квадратных" и "прямоугольных" деталей найдите ширину и высоту стены, которую Саша сможет собрать. Если Саша может собрать идеальную стену разных размеров, перечислите все варианты размеров.

В единственной строке ввода содержатся два целых числа A и B (от 0 до 10^9 включительно, $A + B > 0$): количество "квадратных" и "прямоугольных" деталей соответственно.

В первой строке вывода должно содержаться одно число N -- количество вариантов сборки идеальной стены (возможно, 0). Затем выведите N строк, содержащих по два целых числа (ширина и высота стены) в каждой. Строки должны идти в порядке возрастания ширины.

Пример ввода 1	Пример вывода 1
6 3	3 2 6 3 4 4 3
Пример ввода 2	Пример вывода 2
1 1	0

Стены для первого примера изображены на рисунке.



Во втором случае нельзя собрать стену, удовлетворяющую всем условиям.

Система оценки и описание подзадач

Подзадача 1 (20 баллов)

$1 \leq A+B \leq 1000$, при этом $B=0$ или $A=0$ (присутствуют детали только одного вида)

В этой подзадаче 4 теста, каждый тест оценивается в 5 баллов. Баллы за каждый тест начисляются независимо.

Подзадача 2 (20 баллов)

$1 \leq A \leq 1000$, $B=1$ ("прямоугольная" деталь ровно одна).

В этой подзадаче 4 теста, каждый тест оценивается в 5 баллов. Баллы за каждый тест начисляются независимо.

Некоторые примеры ввода из условия задачи не соответствуют ограничениям подзадачи 1 и 2, для прохождения этих тестов вы должны добавить необходимый код в свое решение, если вы ограничиваетесь только решением подзадач 1 или 2.

Подзадача 3 (30 баллов)

$1 \leq A \leq 1000$, $2 \leq B \leq 1000$.

Необходимые подзадачи: 1, 2.

В этой подзадаче 6 тестов. Баллы за подзадачу начисляются только в случае, если все тесты для этой подзадачи успешно пройдены.

Подзадача 4 (30 баллов)

$1 \leq A \leq 10^9$, $2 \leq B \leq 10^9$.

Необходимые подзадачи: 1, 2, 3.

В этой подзадаче 6 тестов. Баллы за подзадачу начисляются только в случае, если все тесты для этой подзадачи успешно пройдены.

По запросу сообщается результат окончательной проверки на каждом тесте в подзадачах 1 и 2, и о первой ошибке в в подзадачах 3 и 4.

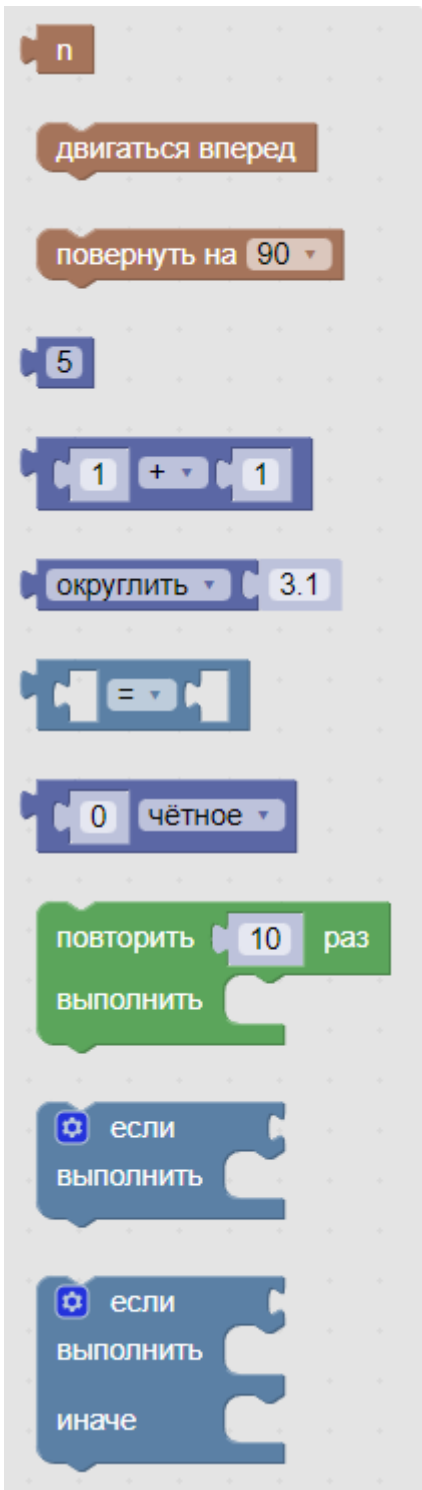
5. Чертежник

Напишите в Blockly программу для робота-чертежника, который может перемещаться только между точками с целыми координатами и поворачивать на углы, кратные 45 градусам. При движении робот проводит линию.

После выполнения программы робот должен вернуться в начальную точку, нарисовав n -угольник, возможно невыпуклый, без самопересечений (робот не должен пересекать нарисованные ранее линии или возвращаться в точку, где ранее был, кроме начальной точки в конце программы).

Форма n -угольника может быть любой, но робот не должен удаляться от начальной точки далее 250 единиц по горизонтали или вертикали.

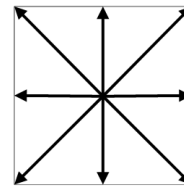
Для написания программы управления роботом вы можете использовать следующие блоки, заданные в категории "Робот":



Первый блок позволяет получить количество углов n -угольника (вы можете задавать это значение в соответствующем поле).

Второй блок указывает роботу двигаться вперед в текущем направлении.

При этом робот перемещается в соседнюю точку с целыми координатами, как показано на рисунке.

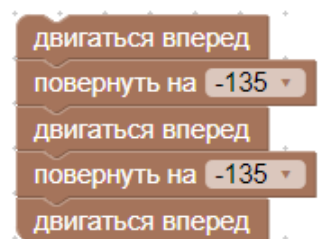


Третий блок указывает роботу изменить направление на указанный угол по часовой стрелке (для положительных значений) или против часовой стрелки (для отрицательных значений).

Первоначально робот направлен направо:



Например, чтобы нарисовать треугольник, нужно выполнить следующие команды:



Следующие пять блоков позволяют выполнять арифметические вычисления, проверять четность и сравнивать значения между собой.

Девятый блок дает возможность повторить выполнение некоторой группы блоков. Например, для рисования шестиугольника можно выполнить программу:



Последние два блока позволяют выбрать выполняемую группу блоков в зависимости от условия. В настройках блока (символ шестеренки) можно добавить или убрать дополнительные ветки выбора. После завершения настройки нужно повторно нажать на символ шестерёнки.

Для создания переменной в категории "Переменные" нужно щелкнуть по кнопке "Создать переменную" и ввести её название.

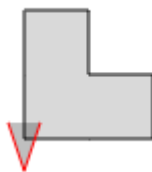
После этого в категории "Переменные" появятся блоки для получения и изменения значения переменной.

Для запуска программы используйте кнопки:



Первая кнопка позволяет отправить ваше решение для проверки в проверяющую систему соревнований, вторая кнопка выполняет запуск программы локально, третья — пошаговое выполнение или временная остановка программы, четвертая — завершение выполнения программы, после которой программа будет выполняться сначала.

Например, после запуска на выполнение программы, показанной выше, робот нарисует шестиугольник:



Две кнопки над полем для рисования позволяют менять масштаб изображения как во время выполнения программы, так и после.

Система оценки и описание подзадач

Подзадача 1 (80 баллов)

Количество углов от 3 до 10.

В этой подзадаче 8 тестов, каждый тест оценивается в 10 баллов. Баллы за каждый тест начисляются независимо.

Подзадача 2 (20 баллов)

Количество углов от 11 до 100.

Необходимые подзадачи: 1.

В этой подзадаче 4 теста, каждый тест оценивается в 5 баллов. Баллы за каждый тест начисляются независимо.

По запросу сообщается результат окончательной проверки на каждом тесте.