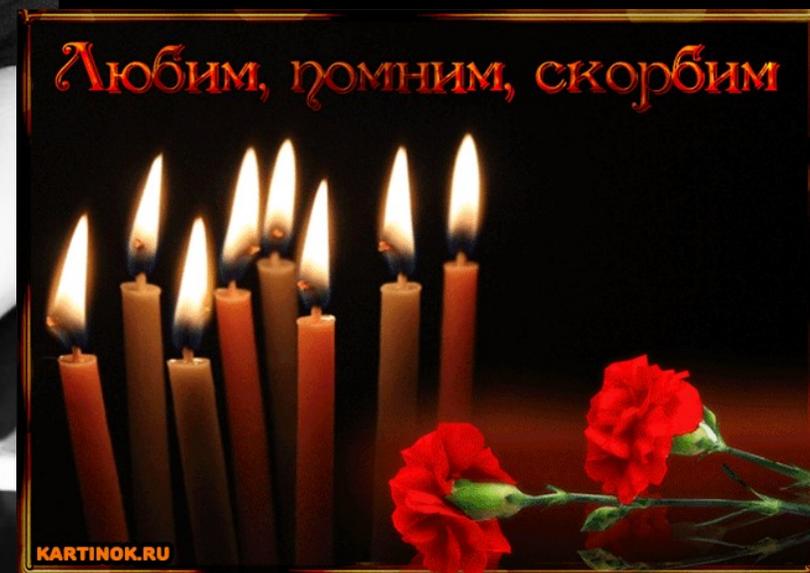


Методика подготовки обучающихся к решению заданий по теме:

Исполнители алгоритмов в среде «Кумир»

Фельдкорен Элеонора Леонардовна,  
учитель информатики  
МОУ «Гимназия г. Раменское»



ИНФОРМАТИКА  
8 класс

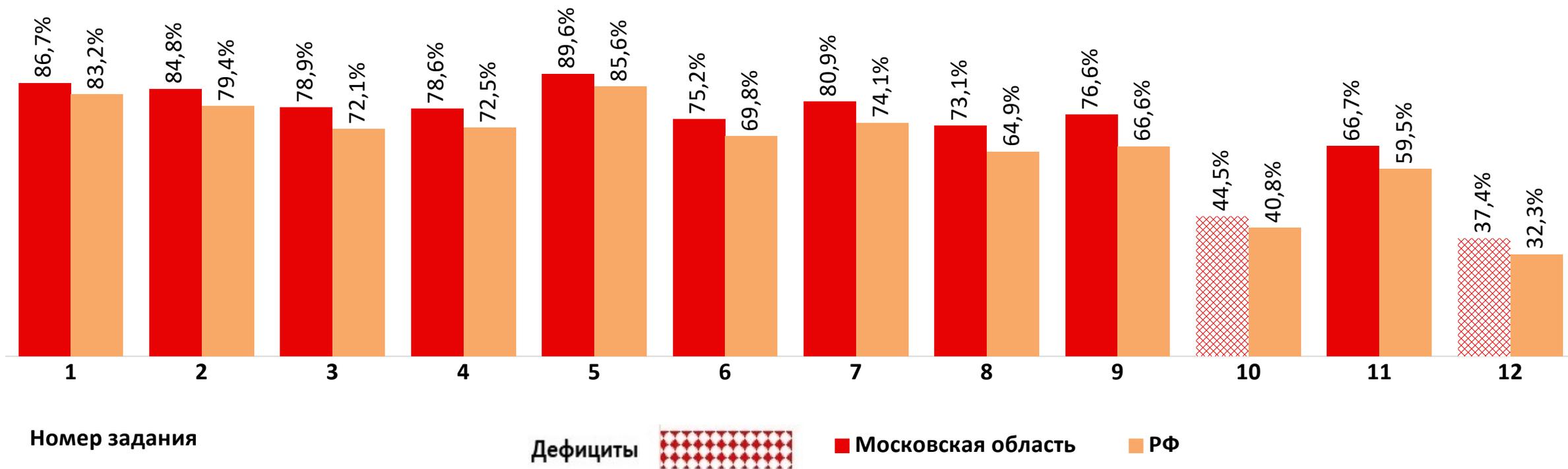
количество участников ВПР  
2024-2025 учебный год

Российская Федерация 129 326

Московская область 9 929

средняя отметка 3,9

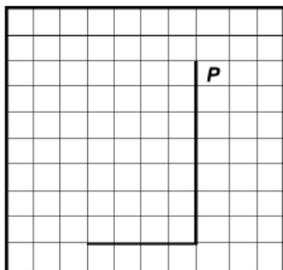
Процент выполнения заданий ВПР в Московской области



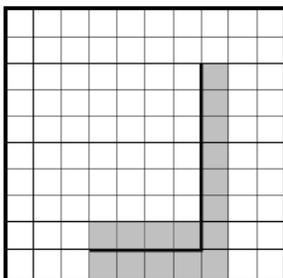
**ИНФОРМАТИКА**  
**8 класс**

Номер задания	Проверяемые требования (умения)	Выявленные дефициты
12	<p>Создавать и выполнять программы для заданного исполнителя «Робот» с использованием циклических алгоритмов.</p> <p>12.1. Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертежник.</p> <p>12.2. Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертежник. Анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений.</p>	<p><i>Предметные знания и умения:</i> создавать и выполнять программы для заданного исполнителя «Робот» с использованием циклических алгоритмов.</p> <p><i>Метапредметные умения:</i></p> <ul style="list-style-type: none"> <li>- устанавливать существенный признак классификации, основания для обобщения и сравнения, критерии проводимого анализа;</li> <li>- самостоятельно формулировать обобщения и выводы по результатам проведенного наблюдения, опыта, исследования; владеть инструментами оценки достоверности полученных выводов и обобщений;</li> <li>- делать выводы с использованием дедуктивных и индуктивных умозаключений, умозаключений по аналогии; формулировать гипотезы о взаимосвязях;</li> <li>- самостоятельно выбирать способ решения учебной задачи (сравнивать несколько вариантов решения, выбирать наиболее подходящий с учетом самостоятельно выделенных критериев);</li> <li>- выявлять и характеризовать существенные признаки объектов (явлений).</li> </ul>

- 12.1 На бесконечном поле имеется вертикальная стена. Длина стены – 7 клеток. От нижнего конца стены влево отходит горизонтальная стена длиной 4 клетки. Робот находится в клетке, расположенной непосредственно справа от верхнего края вертикальной стены. На рисунке указано расположение стен и Робота. Робот обозначен буквой «Р».



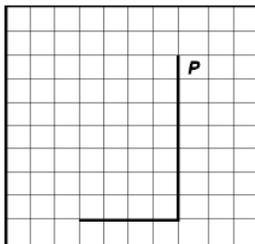
Напишите для Робота программу, использующую 3 циклических алгоритма, закрашивающую все клетки, расположенные непосредственно правее вертикальной стены, ниже горизонтальной стены, угловую клетку и клетки выше горизонтальной стены. Вы можете использовать цикл **нц-раз-кц** или **нц-пока-кц**. Робот должен закрасить только клетки, удовлетворяющие данному условию. На рисунке показаны клетки, которые Робот должен закрасить (см. рисунок).



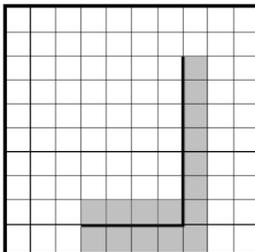
Конечное положение Робота может быть произвольным. При выполнении алгоритма Робот не должен разрушиться. Выполнение алгоритма должно завершиться. Алгоритм может быть выполнен в среде формального исполнителя или записан в текстовом редакторе. Сохраните алгоритм в формате программы Кумир или в текстовом файле. Название файла и каталог для сохранения Вам сообщат организаторы.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	
Команды исполнителя будем записывать жирным шрифтом, а комментарии, поясняющие алгоритм и не являющиеся его частью, – курсивом. Начало комментария будем обозначать символом « ».	
<p><i>  Двигаемся вниз на 7 клеток и закрашиваем все клетки на пути.</i> <b>нц 7 раз</b>     <b>закрасить</b>     <b>вниз</b> <b>кц</b></p> <p><i>  Двигаемся влево на 5 клеток и закрашиваем все клетки на пути.</i> <b>нц 5 раз</b>     <b>закрасить</b>     <b>влево</b> <b>кц</b></p> <p><i>  Обходим стену.</i> <b>вверх</b></p> <p><i>  Двигаемся вправо на 4 клетки и закрашиваем все клетки на пути.</i> <b>нц 4 раз</b>     <b>вправо</b>     <b>закрасить</b> <b>кц</b></p>	<p><i>  Двигаемся вниз, пока не дойдём до конца вертикальной стены, закрашивая все клетки на пути.</i> <b>нц пока не слева свободно</b>     <b>закрасить</b>     <b>вниз</b> <b>кц</b></p> <p><i>  Закрасим угловую клетку и переместимся в начало горизонтальной стены.</i> <b>закрасить</b> <b>влево</b></p> <p><i>  Двигаемся влево до конца горизонтальной стены, закрашивая все клетки на пути.</i> <b>нц пока не сверху свободно</b>     <b>закрасить</b>     <b>влево</b> <b>кц</b></p> <p><i>  Обходим стену.</i> <b>вверх</b></p> <p><i>  Двигаемся вправо до конца горизонтальной стены, закрашивая все клетки на пути.</i> <b>нц пока справа свободно</b>     <b>вправо</b>     <b>закрасить</b> <b>кц</b></p>
Возможны и другие варианты решения	
Указания по оцениванию	Баллы
Алгоритм содержит 3 циклических алгоритма (нц-раз-кц или нц-пока-кц) правильно работает, закрашивает нужные клетки	2
Алгоритм НЕ содержит 3 циклических алгоритма, но правильно работает, закрашивает нужные клетки. ИЛИ Алгоритм содержит 3 циклических алгоритма, правильно работает, но при этом: 1) закрашено не более 5 лишних клеток; 2) остались незакрашенными не более 5 клеток из числа тех, которые должны были быть закрашены	1
Задание выполнено неверно, т.е. не выполнены условия, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл</i>	
	2

- 12.2 На бесконечном поле имеется вертикальная стена. Длина стены неизвестна. От нижнего конца стены влево отходит горизонтальная стена также неизвестной длины. Робот находится в клетке, расположенной непосредственно справа от верхнего края вертикальной стены. На рисунке указан один из возможных способов расположения стен и Робота. Робот обозначен буквой «Р».



Напишите для Робота программу, закрашивающую все клетки, расположенные непосредственно правее вертикальной стены, ниже горизонтальной стены, угловую клетку и клетки выше горизонтальной стены. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок).



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При исполнении алгоритма Робот не должен разрушиться. Выполнение алгоритма должно завершиться.

Алгоритм может быть выполнен в среде формального исполнителя или записан в текстовом редакторе.

Сохраните алгоритм в формате программы Кумир или в текстовом файле. Название файла и каталог для сохранения Вам сообщает организаторы.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
Команды исполнителя будем записывать жирным шрифтом, а комментарии, поясняющие алгоритм и не являющиеся его частью, – курсивом. Начало комментария будем обозначать символом « ». <i>  Двигаемся вниз, пока не дойдём до конца вертикальной стены, закрашивая все клетки на пути.</i> <b>нц пока не слева свободно</b> <b>закрасить</b> <b>вниз</b> <b>кц</b>

<i>  Закрасим угловую клетку и переместимся в начало горизонтальной стены.</i> <b>закрасить</b> <b>влево</b> <i>  Двигаемся влево до конца горизонтальной стены, закрашивая все клетки на пути.</i> <b>нц пока не сверху свободно</b> <b>закрасить</b> <b>влево</b> <b>кц</b> <i>  Обходим стену.</i> <b>вверх</b> <i>  Двигаемся вправо до конца горизонтальной стены, закрашивая все клетки на пути.</i> <b>нц пока справа свободно</b> <b>вправо</b> <b>закрасить</b> <b>кц</b>
Возможны и другие варианты решения

Указания по оцениванию	Баллы
Алгоритм правильно работает при всех допустимых исходных данных	4
При всех допустимых исходных данных верно следующее: 1) выполнение алгоритма завершается, и при этом Робот не разбивается; 2) закрашено не более 10 лишних клеток; 3) остались незакрашенными не более 10 клеток из числа тех, которые должны были быть закрашены	3
Алгоритм содержит 3 циклических алгоритма (нц-раз-кц или нц-пока-кц) правильно работает, закрашивает нужные клетки для обстановки, описанной в задании 12.1.	2
Алгоритм НЕ содержит 3 циклических алгоритма, но правильно работает для обстановки, описанной в задании 12.1. ИЛИ Алгоритм содержит 3 циклических алгоритма, правильно работает для обстановки, описанной в задании 12.1, но при этом: 1) закрашено не более 5 лишних клеток; 2) остались незакрашенными не более 5 клеток из числа тех, которые должны были быть закрашены	1
Задание выполнено неверно, т. е. не выполнены условия, позволяющие поставить 4, 3, 2 или 1 балл	0
<i>Максимальный балл</i>	4

Критерии оценивания задания № 15	Баллы	Количество и доля участников, получивших данный балл	
Алгоритм правильно работает при всех допустимых исходных данных	2	11635	27,00%
При всех допустимых исходных данных верно следующее: 1) выполнение алгоритма завершается, и при этом Робот не разбивается; 2) закрашено не более 10 лишних клеток; 3) остались не закрашенными не более 10 клеток из числа тех, которые должны были быть закрашены.	1	956	2,22%
Задание выполнено неверно, т.е. не выполнены условия, позволяющие поставить 1 или 2 балла	0	30499	70,78%

## МЕСТО РАЗДЕЛА В КОДИФИКАТОРЕ ПРОВЕРЯЕМЫХ ТРЕБОВАНИЙ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ООП ООО

2.1

Раскрывать смысл понятий "исполнитель", "алгоритм", "программа", понимая разницу между употреблением этих терминов в обыденной речи и в информатике

2.2

Описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы

2.3

Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями

2.6

Анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений

3.1

Разбиение задачи на подзадачи. Составление алгоритмов и программ с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителем Робот или другими исполнителями, такими как Черепашка, Чертежник и другими. Выполнение алгоритмов вручную и на компьютере

## ПРОВЕРЯЕМЫЕ ЭЛЕМЕНТЫ СОДЕРЖАНИЯ ООП ООО

- 

Понятие алгоритма. Исполнители алгоритмов. Алгоритм как план управления исполнителем  
Свойства алгоритма. Способы записи алгоритма (словесный, в виде блок-схемы, программа)
- 

Алгоритмические конструкции. Конструкция "следование". Линейный алгоритм. Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных
- 

Конструкция "ветвление": полная и неполная формы. Выполнение и невыполнение условия (истинность и ложность высказывания). Простые и составные условия
- 

Конструкция "повторение": циклы с заданным числом повторений, с условием выполнения, с переменной цикла
- 

Разработка для формального исполнителя алгоритма, приводящего к требуемому результату при конкретных исходных данных. Разработка несложных алгоритмов с использованием циклов и ветвлений для управления формальными исполнителями. Выполнение алгоритмов вручную и на компьютере. Синтаксические и логические ошибки. Отказы

# ПЕРЕЧЕНЬ ЭЛЕМЕНТОВ СОДЕРЖАНИЯ, ПРОВЕРЯЕМЫХ НА ОСНОВНОМ ГОСУДАРСТВЕННОМ ЭКЗАМЕНЕ ПО ИНФОРМАТИКЕ

Кодификатор ОГЭ 2026 г.

ИНФОРМАТИКА, 9 класс. 12 / 16

Код	Проверяемый элемент содержания	В программе какого класса изучается	Наличие данного элемента содержания в кодификаторе ОГЭ прошлых лет
3	<i>Алгоритмы и программирование</i>		
3.1	Свойства алгоритма. Способы записи алгоритма (словесный, в виде блок-схемы, программа). Составление алгоритмов и программ с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителем Робот или другими исполнителями, такими как Черепашка, Чертёжник и другими. Выполнение алгоритмов вручную и на компьютере	9	+

## РАСПРЕДЕЛЕНИЕ ЗАДАНИЙ В ОБОБЩЁННОМ ПЛАНЕ ВАРИАНТА КИМ ОГЭ 2026 ГОДА ПО ИНФОРМАТИКЕ (СПЕЦИФИКАЦИЯ КИМ ОГЭ 2026 Г.)

№	ТРЕБОВАНИЯ К ПРЕДМЕТНЫМ РЕЗУЛЬТАТАМ ОСВОЕНИЯ ООП ООУ	Код КЭС/КТ	Уровень сложности	Максимальный балл за выполнение задания	Примерное время выполнения задания
<b>15</b>	<b>Создавать и выполнять программы для заданного исполнителя</b>	3.1/2.5	<b>Высокий</b>	<b>2</b>	<b>25</b>

## РАСПРЕДЕЛЕНИЕ ЧАСОВ В КАЛЕНДАРНО-ТЕМАТИЧЕСКОМ ПЛАНЕ

10

общее число часов  
в 8 классе на изучение раздела  
«Исполнители и алгоритмы.  
Алгоритмические конструкции»

-  Понятие алгоритма. Исполнители алгоритмов. Алгоритм как план управления исполнителем. Свойства алгоритма.
-  Способы записи алгоритма (словесный, в виде блок-схемы, программа).
-  Алгоритмические конструкции. Конструкция «следование». Линейный алгоритм. Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных.
-  Конструкция «ветвление»: полная и неполная формы. Выполнение и невыполнение условия (истинность и ложность высказывания). Простые и составные условия.
-  Конструкция «повторения»: циклы с заданным числом повторений, с условием выполнения, с переменной цикла.
-  Разработка для формального исполнителя алгоритма, приводящего к требуемому результату при конкретных исходных данных
-  Разработка несложных алгоритмов с использованием циклов и ветвлений для управления формальными исполнителями, такими как Робот, Черепашка, Чертежник.
-  Выполнение алгоритмов вручную и на компьютере. Синтаксические и логические ошибки. Отказы

# РАСПРЕДЕЛЕНИЕ ЧАСОВ В КАЛЕНДАРНО-ТЕМАТИЧЕСКОМ ПЛАНЕ

# 6

общее число часов  
в 9 классе на изучение раздела  
«Разработка алгоритмов и  
программ»



Разбиение задачи на подзадачи.



Составление алгоритмов и программ с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителем Робот или другими исполнителями, такими как Черепашка, Чертежник и др.

## ВИДЫ ДЕЯТЕЛЬНОСТИ

Федеральная рабочая программа | Информатика. 7–9 классы (базовый уровень)



Лекция



Конспект



Решение заданий на анализ и выполнение алгоритмов



Решение заданий на составление алгоритмов



Решение задач ОГЭ (5, 15)

программное содержание	Основные виды деятельности обучающихся
------------------------	--

***Практические работы:***

- 1. Создание и выполнение на компьютере несложных алгоритмов с использованием циклов и ветвлений для управления исполнителями, такими как Робот, Черепашка, Чертежник.*
- 2. Преобразование алгоритма из одной формы записи в другую.*
- 3. Разработка для формального исполнителя алгоритма, приводящего к требуемому результату при конкретных исходных данных.*
- 4. «Ручное» исполнение готовых алгоритмов при конкретных исходных данных*

## Последовательное построение алгоритма

• Метод разработки алгоритма «Сверху вниз» – нисходящий метод или метод пошаговой детализации.

1. Считаем, что перед нами совершенный исполнитель, который сам всё знает и умеет. Поэтому достаточно определить исходные данные и указать результаты алгоритма, а сам алгоритм представить в виде единого предписания – постановки задачи. (можно на этом этапе построить блок-схему).
2. Если исполнитель не обучен для выполнения этого предписания, то нужно представить его в виде совокупности более простых предписаний – разбить задачу на несколько более простых частей.
3. Решение каждой части задачи сформулировать в виде отдельной команды.
4. При наличии в алгоритме предписаний, выходящих за пределы возможностей исполнителя (его системы команд), такие предписания вновь разбиваются на более простые команды.



- Процесс продолжается до тех пор, пока все предписания не будут понятны исполнителю.

Рассмотрим примеры исполнителей.

### Пример 6

Исполнитель **Черепаха** перемещается на экране компьютера, оставляя след в виде линии. Система команд **Черепахи** состоит из следующих команд:

**Вперёд  $n$**  (где  $n$  — целое число) — вызывает передвижение **Черепахи** на  $n$  шагов в направлении движения — в том направлении, куда развёрнута её голова и корпус;



**Направо  $m$**  (где  $m$  — целое число) — вызывает изменение направления движения **Черепахи** на  $m$  градусов по часовой стрелке.

**Запись**

Повтори  $k$  [**Команда 1**] <**Команда 2**> ... <**Команда  $n$** >]

означает, что последовательность команд в скобках повторится  $k$  раз.

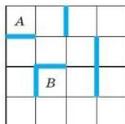
Подумайте, какая фигура появится на экране после выполнения **Черепахой** следующего алгоритма:

Повтори 12 [Направо 45 Вперёд 20 Направо 45]

### Пример 8

Исполнитель **Робот** действует на клетчатом поле, между соседними клетками которого могут стоять стены. Робот передвигается по клеткам поля и может выполнять следующие команды, которым присвоены номера:

1. вверх
2. вниз
3. вправо
4. влево



При выполнении каждой такой команды **Робот** перемещается в соседнюю клетку в указанном направлении. Если же в этом направлении между клетками стоит стена, то **Робот** разрушается.

Что произойдёт с **Роботом** из примера 8, если он выполнит последовательность команд 32323 (здесь цифры обозначают номера команд), начав движение из клетки **A**? Какову последовательность команд следует выполнить **Роботу**, чтобы переместиться из клетки **A** в клетку **B**, не разрушившись при этом от столкновения со стеной?

### Пример 7

Система команд исполнителя **Вычислитель** состоит из двух команд, которым присвоены номера:

1. вычти 1
2. умножь на 3

Первая из них уменьшает число на 1, вторая увеличивает число в 3 раза. При записи алгоритмов для краткости указываются лишь номера команд.

Например, алгоритм 21212 означает следующую последовательность команд:

- умножь на 3
- вычти 1
- умножь на 3
- вычти 1
- умножь на 3

С помощью этого алгоритма число 1 будет преобразовано в 15:  $((1 \cdot 3 - 1) \cdot 3 - 1) \cdot 3 = 15$ .

### Пример 9

К пятизначному натуральному числу применяется следующий алгоритм:

1. Вычислить сумму первых трёх цифр.
2. Вычислить сумму последних двух цифр.
3. Записать полученные два числа друг за другом в порядке возрастания (неубывания).

Пример работы алгоритма для числа 56789:

1.  $5 + 6 + 7 = 18$ .
2.  $8 + 9 = 17$ .

3. Упорядочив, получаем 1718.

Выясним наименьшее и наибольшее пятизначные числа, в результате применения к которым этого алгоритма получается такой же результат.

## Алгоритмы и исполнители §3.1

20. В какой клетке (**A** или **B**) должен находиться исполнитель **Робот** из примера 8, чтобы после выполнения алгоритма 3241 (где цифры — это номера команд **Робота**) в неё же и вернуться?

По ссылке <http://gotourl.ru/11950> вы можете скачать систему программирования **КуМир (Комплект учебных Миров)** со встроенными исполнителями **Робот**, **Чертёжник**, **Водолей** и др. **КуМир** работает в операционных системах **Windows** и **Linux**.

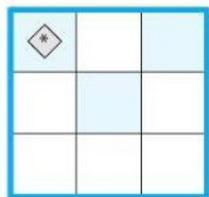
Команда	Описание команды
Вверх	Робот перемещается в соседнюю клетку в указанном направлении.
Вниз	Если в этом направлении между клетками стоит стена, то Робот разрушается
Вправо	
Влево	
Закрасить	Робот закрашивает ту клетку, в которой находится
Сверху свободно	Проверка истинности условия отсутствия стены у соответствующей стороны той клетки, где находится Робот: стены нет — «истина», иначе «ложь»
Снизу свободно	
Слева свободно	
Справа свободно	
Сверху стена	Проверка истинности условия наличия стены у соответствующей стороны той клетки, где находится Робот: стена есть — «истина», иначе «ложь»
Снизу стена	
Слева стена	
Справа стена	
Клетка закрашена	Проверка истинности условия: клетка закрашена — «истина», иначе «ложь»
если <условие> то <команда 1> иначе <команда 2> все	Организация ветвления: если <условие> истинно, то выполняется <команда 1>; если <условие> ложно, то выполняется <команда 2>
нц пока <условие> <последовательность команд> кц	Организация цикла: пока <условие> истинно, выполняется <последовательность команд>

## ПРИМЕР ЗАДАНИЙ, РЕШАЕМЫХ НА УРОКАХ

### 3.4.2. Ограниченность линейных алгоритмов

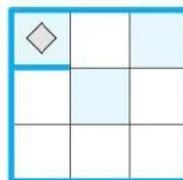
#### Пример 5

У исполнителя Робот есть четыре команды перемещения (вверх, вниз, влево и вправо), при выполнении каждой из них Робот перемещается на одну клетку в соответствующем направлении. По команде закрасить Робот закрашивает клетку, в которой он находится. Запишем линейный алгоритм, исполняя который Робот нарисует на клетчатом поле следующий узор и вернётся в исходное положение, обозначенное звездочкой:

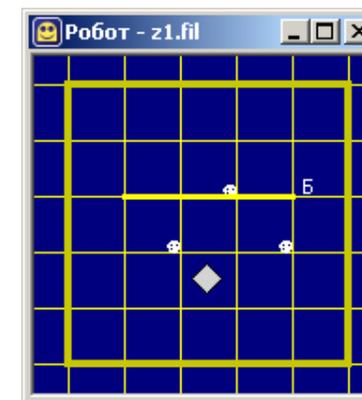


```
алг узор
нач
  закрасить
  вправо
  вправо
  закрасить
  вниз
  влево
  закрасить
  влево
  вверх
кон
```

А теперь подумайте, что произойдёт, если Робот приступит к выполнению этой же программы в обстановке, в которой между двумя клетками есть стена:



Очевидно, что Робот закрасит все клетки, но не сможет выполнить последнюю команду: попытается подняться вверх, Робот столкнётся со стеной, что приведёт к его разрушению.



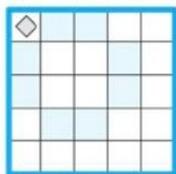
использовать **Робот**  
алг Команды  
нач  
 вверх  
 влево  
 закрасить  
 вправо  
 вправо  
 закрасить  
 вправо  
 вверх  
 влево  
 влево  
 закрасить  
 вправо  
 вправо  
кон

## ПРИМЕР ЗАДАНИЙ, РЕШАЕМЫХ НА УРОКАХ

### Конструкция «ветвление». Разветвляющиеся алгоритмы

§3.5

Запишите линейный алгоритм, исполняя который Робот нарисует на клетчатом поле следующий узор и вернется в исходное положение.



Задание распечатывается  
каждому учащемуся

Слабоуспевающим учащимся  
рекомендуется использовать  
конспект

#### Пример 7

Исполнитель Робот может выполнять ту или иную последовательность действий в зависимости от выполнения следующих простых условий:

справа свободно  
слева свободно  
сверху свободно  
снизу свободно  
клетка чистая

справа стена  
слева стена  
сверху стена  
снизу стена  
клетка закрашена

Подумайте, в какую клетку переместится Робот из клетки, обозначенной ромбиком, при выполнении следующего фрагмента алгоритма.

**если** справа свободно **или** снизу свободно  
**то** закрасить

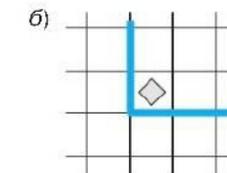
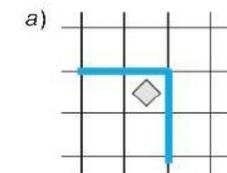
**все**

**если** справа стена  
**то** влево

**все**

**если** слева стена  
**то** вправо

**все**

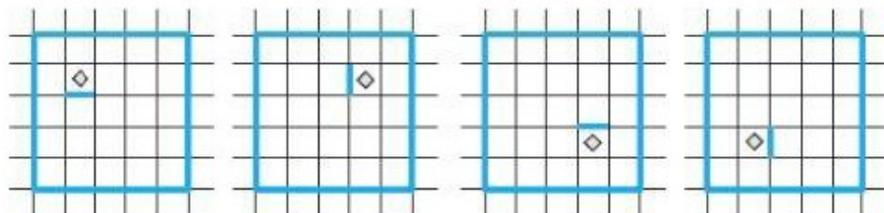


## ПРИМЕР ЗАДАНИЙ, РЕШАЕМЫХ НА УРОКАХ

15. Робот находится внутри поля размером  $5 \times 5$  клеток. Рядом с Роботом есть стена длиной в одну клетку. Составьте в среде (системе программирования) КуМир один из следующих алгоритмов:

- Робот закрашивает две клетки: клетку, в которой он находится в стартовой позиции, и клетку с другой стороны от стены;
- Робот закрашивает клетку с другой стороны стены и возвращается назад;
- Робот закрашивает клетку, в которой он стоит, и «прячется» за стену.

Протестируйте программу, устанавливая начальное положение Робота в клетках, отмеченных ромбом.



Учебник 8-го класса

Задание выполняется за компьютером в среде КуМир, тексты программ записываются в тетради.

Можно варьировать постановку задания учащимся.

Слабоуспевающим учащимся рекомендуется использовать конспект

## ПРИМЕР ЗАДАНИЙ, РЕШАЕМЫХ НА УРОКАХ

### Конструкция «повторение». Циклические алгоритмы §3.6

#### Пример 2

Правее Робота (клетка с ромбиком) расположен коридор неизвестной длины. Необходимо, чтобы Робот закрасил все клетки этого коридора.



Предлагаем построить линейный и разветвляющийся алгоритм. Запускаем дискуссию, какой из них быстрее, по мнению учащихся компактнее, быстрее, удобнее. Делаем вывод... Объясняем тему...

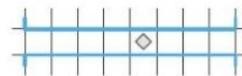
Соответствующий алгоритм для Робота будет иметь вид:

```
нц пока справа свободно
  вправо
  закрась
кц
```

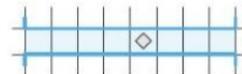
Учебник 8-го класса

### Конструирование алгоритмов §1.1

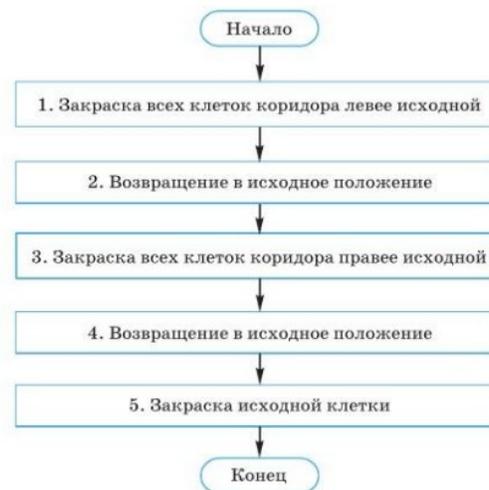
Известно, что Робот находится где-то в горизонтальном коридоре. Ни одна из клеток коридора не закрашена.



Составим алгоритм, под управлением которого Робот закрасит все клетки этого коридора и вернется в исходное положение.



Представим план действий Робота следующими укрупнёнными шагами (модулями):



Учебник 9-го класса

Детализируем каждый из пяти модулей.

```

использовать Робот
алг
нач
  влево
нц пока сверху стена и снизу стена
  закрасить; влево
кц
  вправо
нц пока клетка закрашена
  вправо
кц
  вправо
нц пока сверху стена и снизу стена
  закрасить; вправо
кц
  влево
нц пока клетка закрашена
  влево
кц
  закрасить
кон
  
```

Предлагаем выполнить приведённый выше алгоритм в среде КуМир, предварительно создав в ней стартовую обстановку – горизонтальный коридор.

## ПРИМЕР ЗАДАНИЙ, РЕШАЕМЫХ НА УРОКАХ

Известно, что Робот находится где-то в горизонтальном коридоре. Ни одна из клеток коридора не закрашена.



Составим алгоритм, под управлением которого Робот закрасит все клетки этого коридора и вернётся в исходное положение.



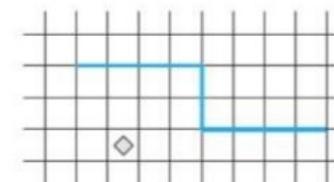
Разработайте алгоритм решения исходной задачи при условии, что в системе команд Робота отсутствуют следующие команды:

- сверху стена
- снизу стена
- слева стена
- справа стена

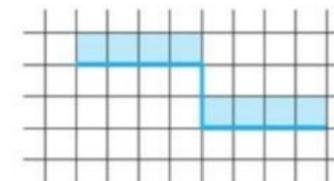
Модифицируйте алгоритм таким образом, чтобы с его помощью можно было бы решить аналогичную задачу в вертикальном коридоре. Убедитесь в работоспособности разработанного алгоритма.

- На бесконечном поле имеется стена, состоящая из двух горизонтальных и одного вертикального участков, примерная форма которой представлена на рисунке. Длины участков неизвестны. Робот находится в клетке под левой горизонтальной частью стены.

На рисунке указан один из возможных вариантов расположения стены и Робота.



Задача Робота — закрасить клетки, примыкающие сверху к горизонтальным участкам стены. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки:



Конечное расположение Робота может быть произвольным. Задача должна быть решена для стены, горизонтальные и вертикальный участки которой могут иметь любые размеры.

## ПРИМЕР ЗАДАНИЯ ОГЭ (15)

15 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. У Робота есть девять команд. Четыре команды – это команды-приказы:

**вверх вниз влево вправо**

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →. Если Робот получил команду передвижения сквозь стену, то он разрушится.

Также у Робота есть команда **закрасить**, при которой закрашивается клетка, где Робот находится в настоящий момент.

Ещё четыре команды – это команды проверки условий. Эти команды проверяют, свободен ли путь для Робота в каждом из четырёх возможных направлений:

**сверху свободно снизу свободно слева свободно справа свободно**

Эти команды можно использовать вместе с условием «если», имеющим следующий вид:

**если условие то**  
*последовательность команд*  
**все**

Здесь *условие* – одна из команд проверки условия.

*Последовательность команд* – это одна или несколько любых команд-приказов.

Например, для передвижения на одну клетку вправо, если справа нет стенки, и закрашивания клетки можно использовать такой алгоритм:

**если справа свободно то**  
**вправо**  
**закрасить**  
**все**

В одном условии можно использовать несколько команд проверки условий, применяя логические связки **и**, **или**, **не**, например:

**если (справа свободно) и (не снизу свободно) то**  
**вправо**  
**все**

Для повторения последовательности команд можно использовать цикл «пока», имеющий следующий вид:

**нц пока условие**  
*последовательность команд*  
**кц**

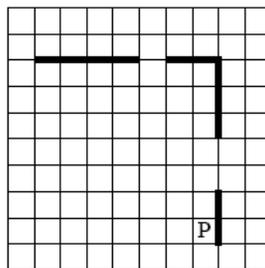
Например, для движения вправо, пока это возможно, можно использовать следующий алгоритм:

**нц пока справа свободно**  
**вправо**  
**кц**

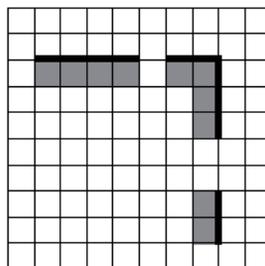
**Выполните задание.**

На бесконечном поле есть горизонтальная и вертикальная стены. Правый конец горизонтальной стены соединён с верхним концом вертикальной стены. Длины стен неизвестны. В каждой стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной рядом с вертикальной стеной слева от её нижнего конца.

На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно левее вертикальной стены и ниже горизонтальной стены. Проходы должны остаться незакрашенными. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок).



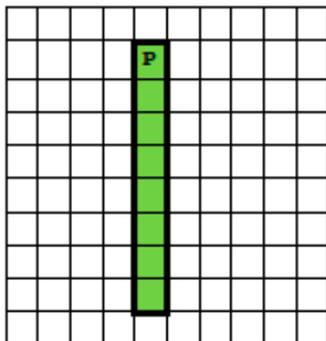
При исполнении алгоритма Робот не должен разрушиться, выполнение алгоритма должно завершиться. Конечное положение Робота может быть произвольным.

Алгоритм должен решать задачу для любого допустимого расположения стен и любого расположения и размера проходов внутри стен.

Алгоритм может быть выполнен в среде формального исполнителя или записан в текстовом редакторе.

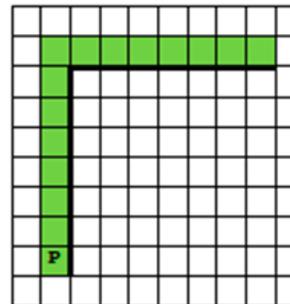
Сохраните алгоритм в формате программы Кумир или в текстовом файле. Название файла и каталог для сохранения Вам сообщат организаторы экзамена.

## ПРИМЕР ЗАДАНИЯ ОГЭ (15)



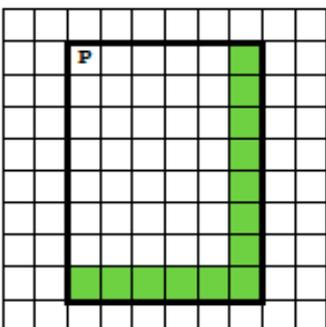
Робот находится в верхней клетке узкого вертикального коридора. Ширина коридора – одна клетка, длина коридора может быть произвольной.

Напишите для Робота алгоритм, закрашивающий все клетки внутри коридора и возвращающий Робота в исходную позицию.



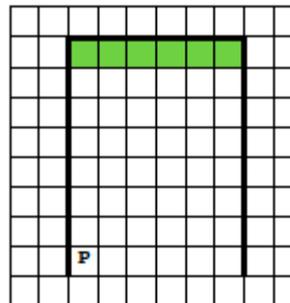
На бесконечном поле имеется вертикальная стена. Длина стены неизвестна. От верхнего конца стены вправо отходит горизонтальная стена также неизвестной длины. Робот находится в клетке, расположенной слева от нижнего края вертикальной стены.

Напишите для Робота алгоритм, закрашивающий все клетки, расположенные левее вертикальной стены и выше горизонтальной стены и прилегающие к ним.



На бесконечном поле имеются 4 стены, расположенные в форме прямоугольника. Длины вертикальных и горизонтальных стен неизвестны. Робот находится в клетке, расположенной в левом верхнем углу прямоугольника.

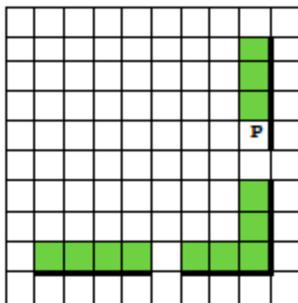
Напишите для Робота алгоритм, закрашивающий все клетки, расположенные с внутренней стороны правой и нижней стен.



На бесконечном поле имеются две вертикальные стены одинаковой длины, расположенные точно одна напротив другой. Длина стен неизвестна. Расстояние между стенами неизвестно. Робот находится справа от первой стены в клетке, расположенной у её нижнего края.

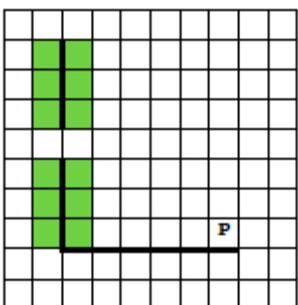
Напишите для Робота алгоритм, закрашивающий все клетки самого верхнего ряда, расположенные между стенами.

## ЗАДАНИЯ ДЛЯ ТРЕНИРОВКИ



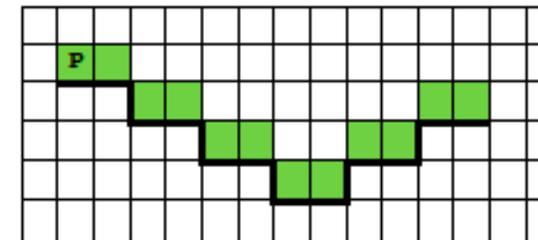
На бесконечном поле есть горизонтальная и вертикальная стены. Правый конец горизонтальной стены соединён с нижним концом вертикальной стены. **Длины стен неизвестны.** В каждой стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно слева от вертикальной стеной у верхнего конца прохода.

Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно выше горизонтальной стены и левее вертикальной стены, кроме клетки, в которой находится Робот перед выполнением программы.



На бесконечном поле есть горизонтальная и вертикальная стены. Левый конец горизонтальной стены соединён с нижним концом вертикальной стены. **Длины стен неизвестны.** В вертикальной стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно над горизонтальной стеной у её правого конца.

Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно левее и правее вертикальной стены. Проход должен остаться незакрашенным.



На бесконечном поле имеется лестница. Сначала лестница слева направо спускается вниз, затем поднимается вверх. Высота каждой ступени – одна клетка, ширина – две клетки. Робот находится на первой ступеньке лестницы, в левой клетке. Количество ступеней, ведущих вниз, и количество ступеней, ведущих вверх, неизвестно.

Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно над ступенями лестницы