

**Описание  
проверочной работы по информатике  
для обучающихся 8-х классов  
образовательных организаций города Москвы**

**1. Назначение проверочной работы**

Проверочная работа проводится с целью осуществления мониторинга уровня и качества подготовки обучающихся в порядке, принятом Департаментом образования и науки города Москвы.

Назначение проверочной работы по учебному предмету «Информатика» – оценить качество общеобразовательной подготовки обучающихся 8-х классов в соответствии с требованиями федерального государственного образовательного стандарта основного общего образования и федеральной образовательной программы основного общего образования.

Период проведения – май 2025 года.

**2. Документы, определяющие содержание и характеристики проверочной работы**

Содержание и основные характеристики проверочной работы определяются на основе следующих документов:

– Федеральный государственный образовательный стандарт основного общего образования (утверждён приказом Минобрнауки России от 17.12.2010 № 1897);

– Федеральный государственный образовательный стандарт основного общего образования (утверждён приказом Минпросвещения России от 31.05.2021 № 287);

– Федеральная образовательная программа основного общего образования (утверждена приказом Минпросвещения России от 18.05.2023 № 370);

– Федеральный перечень учебников, допущенных к использованию при реализации имеющих государственную аккредитацию образовательных программ начального общего, основного общего, среднего общего образования организациями, осуществляющими образовательную деятельность (утверждён приказом Минпросвещения России от 21.09.2022 № 858);

– Универсальный кодификатор распределённых по классам проверяемых требований к результатам освоения основной образовательной программы основного общего образования и элементов содержания по информатике (подготовлен ФГБНУ «ФИПИ»).

**3. Условия проведения проверочной работы**

При организации и проведении работы необходимо строгое соблюдение порядка организации и проведения независимой диагностики.

Проверочная работа проводится в компьютерной форме.

Дополнительные материалы и оборудование: непрограммируемый калькулятор. При выполнении заданий на протяжении всей диагностики необходимо обеспечить доступ к учебной среде исполнителя «Робот».

**4. Время выполнения проверочной работы**

Время выполнения проверочной работы – 45 минут без учёта времени на перерыв для разминки глаз. В работе предусмотрен один автоматический пятиминутный перерыв.

**5. Содержание и структура проверочной работы**

Каждый вариант проверочной работы состоит из 8 заданий.

Проверочная работа содержит задания, направленные на проверку различных блоков умений, формируемых при изучении курса информатики.

Распределение заданий по разделам содержания учебного предмета представлено в таблице 1.

*Таблица 1*

№ п/п	Разделы содержания учебного предмета	Количество заданий
1.	Теоретические основы информатики	7
2.	Алгоритмы и программирование	5

В таблице 2 приведён перечень планируемых результатов обучения.

*Таблица 2*

№ п/п	Контролируемые требования (КТ) к уровню подготовки обучающихся (умения)
1.	Записывать и сравнивать целые числа от 0 до 1024 в различных позиционных системах счисления (с основаниями 2, 8, 16), выполнять арифметические операции над ними
2.	Записывать логические выражения с использованием дизъюнкции, конъюнкции и отрицания; определять истинность логических выражений, если известны значения истинности входящих в него переменных; строить таблицы истинности для логических выражений
3.	Описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы
4.	Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертёжник

**Обобщённый план  
проверочной работы по информатике  
для обучающихся 8-х классов  
образовательных организаций города Москвы**

Используются следующие условные обозначения:  
Б – базовый уровень сложности, П – повышенный уровень сложности.

5.	Анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений
6.	Создавать и отлаживать программы на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык), реализующие несложные алгоритмы обработки числовых данных с использованием циклов и ветвлений, в том числе реализующие проверку делимости одного целого числа на другое, проверку натурального числа на простоту, выделения цифр из натурального числа

**6. Порядок оценивания выполнения отдельных заданий и работы в целом**

Верное выполнение каждого из заданий 1–5 и 7 оценивается 1 баллом; заданий 6 и 8 оценивается 2 баллами.

Максимальный балл за выполнение всей проверочной работы – 10 баллов.

В приложении 1 приведён обобщённый план проверочной работы.

На сайте ГАОУ ДПО МЦКО <http://demo.mcko.ru/test/> размещены образцы заданий в компьютерной форме, примерные типы и форматы которых могут быть представлены в отдельных вариантах проверочной работы.

В приложении 2 приведены ответы и указания к оцениванию образцов заданий проверочной работы, представленных на сайте ГАОУ ДПО МЦКО.

№ задания	Проверяемые элементы содержания	Код ПЭС	Проверяемые требования к результатам освоения основной образовательной программы	Код ПРО	Уровень сложности	Макс. балл
1	Двоичная система счисления. Перевод натуральных чисел в двоичную систему счисления. Восьмеричная система счисления. Перевод чисел из восьмеричной системы в двоичную и десятичную системы и обратно. Шестнадцатеричная система счисления. Перевод чисел из шестнадцатеричной системы в двоичную, восьмеричную и десятичную системы и обратно. Позиционные и непозиционные системы счисления. Алфавит. Основание. Развёрнутая форма записи числа. Перевод в десятичную систему чисел, записанных в других системах счисления	1.3	Записывать и сравнивать целые числа от 0 до 1024 в различных позиционных системах счисления (с основаниями 2, 8, 16), выполнять арифметические операции над ними	1.1; 1.2	П	1
2	Арифметические операции в двоичной системе счисления	1.4	Записывать и сравнивать целые числа от 0 до 1024 в различных позиционных системах счисления	1.2	Б	1

Настоящий текст является объектом авторского права. Свободное и безвозмездное использование любых материалов, входящих в состав данного текста, ограничено использованием в личных целях и допускается исключительно в некоммерческих целях. Нарушение вышеуказанных положений является нарушением авторских прав и влечёт наступление гражданской, административной и уголовной ответственности в соответствии с законодательством Российской Федерации. В случае самостоятельного использования материалов теста ГАОУ ДПО МЦКО не несёт ответственности за утрату актуальности текста.

Настоящий текст является объектом авторского права. Свободное и безвозмездное использование любых материалов, входящих в состав данного текста, ограничено использованием в личных целях и допускается исключительно в некоммерческих целях. Нарушение вышеуказанных положений является нарушением авторских прав и влечёт наступление гражданской, административной и уголовной ответственности в соответствии с законодательством Российской Федерации. В случае самостоятельного использования материалов теста ГАОУ ДПО МЦКО не несёт ответственности за утрату актуальности текста.

			(с основаниями 2, 8, 16), выполнять арифметические операции над ними			
3	Логические выражения. Правила записи логических выражений. Построение таблиц истинности логических выражений.	1.6	Записывать логические выражения с использованием дизъюнкции, конъюнкции и отрицания; определять истинность логических выражений, если известны значения истинности входящих в него переменных; строить таблицы истинности для логических выражений	1.4	Б	1
4	Алгоритмические конструкции. Конструкция «следование». Линейный алгоритм. Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных. Конструкция «повторение»: циклы с заданным числом повторений, с условием выполнения, с переменной цикла	2.3 2.5	Описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы. Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертежник.	2.2 2.3	Б	1
5	Ветвления. Составные условия (запись логических выражений на изучаемом языке программирования). Нахождение минимума и максимума из двух, трех и четырех чисел. Решение квадратного уравнения, имеющего вещественные корни.	2.10	Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертежник	2.6	Б	1
6	Логические выражения. Правила записи	1.6	Записывать логические выражения с	1.4	П	2

Настоящий текст является объектом авторского права. Свободное и безвозмездное использование любых материалов, входящих в состав данного текста, ограничено использованием в личных целях и допускается исключительно в некоммерческих целях. Нарушение вышеуказанных положений является нарушением авторских прав и влечёт наступление гражданской, административной и уголовной ответственности в соответствии с законодательством Российской Федерации. В случае самостоятельного использования материалов теста ГАОУ ДПО ЦИКО не несёт ответственности за утрату актуальности текста.

© Московский центр качества образования.

	логических выражений. Построение таблиц истинности логических выражений		использованием дизъюнкции, конъюнкции и отрицания; определять истинность логических выражений, если известны значения истинности входящих в него переменных; строить таблицы истинности для логических выражений			
7	Конструкция «повторение»: циклы с заданным числом повторений, с условием выполнения, с переменной цикла	2.5	Составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертежник.	2.3	Б	1
8	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	2.6	Создавать и отлаживать программы на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык), реализующие несложные алгоритмы обработки числовых данных с использованием циклов и ветвлений, в том числе реализующие проверку делимости одного целого числа на другое, проверку натурального числа на простоту, выделения цифр из натурального числа	2.3; 2.6	П	2

Настоящий текст является объектом авторского права. Свободное и безвозмездное использование любых материалов, входящих в состав данного текста, ограничено использованием в личных целях и допускается исключительно в некоммерческих целях. Нарушение вышеуказанных положений является нарушением авторских прав и влечёт наступление гражданской, административной и уголовной ответственности в соответствии с законодательством Российской Федерации. В случае самостоятельного использования материалов теста ГАОУ ДПО ЦИКО не несёт ответственности за утрату актуальности текста.

© Московский центр качества образования.

**Ответы и указания к оцениванию образцов заданий проверочной работы по информатике  
для обучающихся 8-х классов образовательных организаций города Москвы**

№ задания	Ответ (эталон)	Макс. балл	Указания к оцениванию	Балл																																																																				
1	<div style="text-align: right; margin-bottom: 5px;"><span style="border: 1px solid black; padding: 2px;">Справочный материал</span></div> <p>Переведите десятичное число 78 в восьмеричную систему счисления. Основание системы счисления писать не нужно.</p> <p>Ответ: <input style="width: 150px;" type="text" value="116"/>.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p align="center"><b>Справочный материал</b> <span style="float: right;">✕</span></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Восьмеричная цифра</th> <th>Двоичная триада</th> <th>Шестнадцатеричная цифра</th> <th>Двоичная тетрада</th> </tr> </thead> <tbody> <tr><td>0</td><td>000</td><td>0</td><td>0000</td></tr> <tr><td>1</td><td>001</td><td>1</td><td>0001</td></tr> <tr><td>2</td><td>010</td><td>2</td><td>0010</td></tr> <tr><td>3</td><td>011</td><td>3</td><td>0011</td></tr> <tr><td>4</td><td>100</td><td>4</td><td>0100</td></tr> <tr><td>5</td><td>101</td><td>5</td><td>0101</td></tr> <tr><td>6</td><td>110</td><td>6</td><td>0110</td></tr> <tr><td>7</td><td>111</td><td>7</td><td>0111</td></tr> <tr><td></td><td></td><td>8</td><td>1000</td></tr> <tr><td></td><td></td><td>9</td><td>1001</td></tr> <tr><td></td><td></td><td>A</td><td>1010</td></tr> <tr><td></td><td></td><td>B</td><td>1011</td></tr> <tr><td></td><td></td><td>C</td><td>1100</td></tr> <tr><td></td><td></td><td>D</td><td>1101</td></tr> <tr><td></td><td></td><td>E</td><td>1110</td></tr> <tr><td></td><td></td><td>F</td><td>1111</td></tr> </tbody> </table> <p align="right" style="margin-top: 10px;"><span style="border: 1px solid black; padding: 2px;">Ok</span></p> </div> <p><b>ИЛИ</b></p>	Восьмеричная цифра	Двоичная триада	Шестнадцатеричная цифра	Двоичная тетрада	0	000	0	0000	1	001	1	0001	2	010	2	0010	3	011	3	0011	4	100	4	0100	5	101	5	0101	6	110	6	0110	7	111	7	0111			8	1000			9	1001			A	1010			B	1011			C	1100			D	1101			E	1110			F	1111	1	<p>Ответ совпадает с эталоном.</p>	1
Восьмеричная цифра	Двоичная триада	Шестнадцатеричная цифра	Двоичная тетрада																																																																					
0	000	0	0000																																																																					
1	001	1	0001																																																																					
2	010	2	0010																																																																					
3	011	3	0011																																																																					
4	100	4	0100																																																																					
5	101	5	0101																																																																					
6	110	6	0110																																																																					
7	111	7	0111																																																																					
		8	1000																																																																					
		9	1001																																																																					
		A	1010																																																																					
		B	1011																																																																					
		C	1100																																																																					
		D	1101																																																																					
		E	1110																																																																					
		F	1111																																																																					
			Другие варианты.	0																																																																				

	<p style="text-align: right;"><a href="#">Справочный материал</a></p> <p>Какое из чисел <math>a</math>, записанных в двоичной системе, удовлетворяет условию <math>B_{2_{16}} &lt; a &lt; 264_8</math>?</p> <p> <input type="radio"/> 10110001    <input checked="" type="radio"/> 10110011    <input type="radio"/> 10110101    <input type="radio"/> 10100010 </p>			
2	<p style="text-align: right;"><a href="#">Справочный материал</a></p> <p>Выполните сложение:</p> <p style="text-align: center;"><math>345_8 + 162_8</math></p> <p>Ответ запишите в восьмеричной системе счисления. Основание системы счисления записывать не нужно.</p> <p>Ответ: <input type="text" value="527"/>.</p> <p><b>ИЛИ</b></p> <p>Выполните вычитание:</p> <p style="text-align: center;"><math>101110_2 - 1011_2</math></p> <p>Ответ запишите в двоичной системе счисления. Основание системы счисления записывать не нужно.</p> <p>Ответ: <input type="text" value="100011"/>.</p>	1	<p>Ответ совпадает с эталоном.</p>	1
			<p>Другие варианты.</p>	0

3

Справочный материал

Укажите имя, для которого **ЛОЖНО** высказывание.

**НЕ** (первая буква гласная) **ИЛИ** (последняя буква гласная).

Анна
  Максим
  Татьяна
  Олег

Справочный материал

Обозначения для логических связок (операций):

- а) отрицание (инверсия, логическое НЕ) обозначается  $\neg$  (например,  $\neg A$ );
- б) конъюнкция (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );
- в) дизъюнкция (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );
- г) символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).

Приоритеты логических операций: отрицание (инверсия), конъюнкция (логическое умножение, логическое И), дизъюнкция (логическое сложение, логическое ИЛИ).

Таким образом,  $\neg A \wedge B \vee C \wedge D$  обозначает  $((\neg A) \wedge B) \vee (C \wedge D)$ . Возможна запись  $A \wedge B \wedge C$  вместо  $(A \wedge B) \wedge C$ . То же относится и к дизъюнкции: возможна запись  $A \vee B \vee C$  вместо  $(A \vee B) \vee C$ .

**ИЛИ**

Справочный материал

Заполните недостающую часть таблицы истинности следующего выражения:

$$A \vee \neg B$$

A	B	$\neg B$	$A \vee \neg B$
0	0	1	1
0	1	0	0
1	0	1	1
1	1	0	1

1

Ответ совпадает с эталоном.

1

Другие варианты.

0

4

У исполнителя Удвоитель две команды, которым присвоены номера:

- 1. вычти 1
- 2. умножь на 2

Первая из них уменьшает число на экране на 1, вторая удваивает его.

Составьте алгоритм получения из числа 5 числа 30, содержащий не более 5 команд. В ответе запишите только номера команд в соответствующей алгоритму последовательности. Если таких алгоритмов более одного, то запишите любой из них.

Например, 12221 – это алгоритм:

вычти 1  
 умножь на 2  
 умножь на 2  
 умножь на 2  
 вычти 1,  
 который преобразует число 4 в число 23.

Ответ:  .

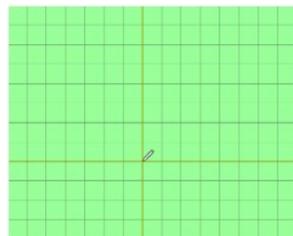
### ИЛИ

Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду:

**Сместиться на (a, b)** (где a, b – целые числа), перемещающую Чертёжника из точки с координатами (x, y) в точку с координатами (x + a, y + b).  
 Если числа a, b положительные, значение соответствующей координаты увеличивается; если отрицательные, значение уменьшается.

Например, если Чертёжник находится в точке с координатами (1, 2), то команда **Сместиться на (3, -3)** переместит Чертёжника в точку (4, -1).

Запись  
**Повтори k раз**  
**Команда1 Команда2 Команда3**  
**Конец**  
 означает, что последовательность команд **Команда1**  
**Команда2 Команда3** повторится k раз.



Начальное положение Чертёжника

Чертёжнику был дан для исполнения следующий алгоритм:

**Повтори 2 раз**  
**Сместиться на (1, 3) Сместиться на (1, -2)**  
**Конец**  
**Сместиться на (2, 6)**

На какую одну команду можно заменить этот алгоритм, чтобы Чертёжник оказался в той же точке, что и после выполнения алгоритма?

- Сместиться на (4, 7)
- Сместиться на (-6, -8)
- Сместиться на (6, 8)
- Сместиться на (-4, -7)

1

Ответ совпадает с эталоном.

1

Другие варианты.

0

5

Ниже приведена программа, записанная на четырёх языках программирования.

Python	Паскаль
<pre>s = int(input()) t = int(input()) if (s &lt; 10) or (t &gt; 10):     print("YES") else:     print("NO")</pre>	<pre>var s, t: integer; begin   readln(s);   readln(t);   if (s &lt; 10) or (t &gt; 10)   then     writeln("YES")   else     writeln("NO") end.</pre>
C++	Алгоритмический язык
<pre>#include &lt;iostream&gt; using namespace std; int main(){   int s, t;   cin &gt;&gt; s;   cin &gt;&gt; t;   if (s &lt; 10    t &gt; 10)     cout &lt;&lt; "YES" &lt;&lt; endl;   else     cout &lt;&lt; "NO" &lt;&lt; endl;   return 0; }</pre>	<pre>алг нач   цел s, t   ввод s   ввод t   если s &lt; 10 или t &gt; 10   то вывод "YES"   иначе вывод "NO" все кон</pre>

Было проведено 5 запусков программы, при которых в качестве значений переменных **s** и **t** вводились пары чисел (**s**, **t**).

Выберите те запуски, при которых программа напечатала "NO".

- (15, 9)
  (5, 11)
  (0, 9)
  (3, 11)
  (17, 10)

1

Ответ совпадает с эталоном.

1

Другие варианты.

0

6

Справочный материал

Заполните недостающую часть таблицы истинности следующего выражения:

$$(\neg A \vee B \wedge \neg C) \wedge C$$

A	B	C	$\neg A$	$\neg C$	$B \wedge \neg C$	$\neg A \vee B \wedge \neg C$	$(\neg A \vee B \wedge \neg C) \wedge C$
0	0	0	1	1	0	1	0
0	0	1	1	0	0	1	1
0	1	0	1	1	1	1	0
0	1	1	1	0	0	1	1
1	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0
1	1	0	0	1	1	1	0
1	1	1	0	0	0	0	0

2

Ответы во всех столбцах совпадают с эталоном.

2

Ответы только на столбцы 3, 4, 5, 6 совпадают с эталоном

1

Другие варианты.

0

7

Исполнитель Черепашка перемещается на экране компьютера, оставляя след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

*Вперед (n)* (где  $n$  – целое число), вызывающая передвижение Черепашки на  $n$  шагов в направлении движения.

*Направо (m)* (где  $m$  – целое число), вызывающая изменение направления движения на  $m$  градусов по часовой стрелке.

Запись **Повтори  $k$  [Команда1 Команда2 Команда3]** означает, что последовательность команд в скобках повторится  $k$  раз.

Черепашке был дан для исполнения следующий алгоритм:

**Повтори 7 [Вперед(4) Направо(60)]**

Постройте многоугольник в среде исполнителя «Черепаха» программы Кумир и посчитайте количество точек с целыми координатами, которые находятся внутри фигуры (точки на границе считать не нужно).

Ответ: .

1

Ответ совпадает с эталоном.

1

Другие варианты.

0

8

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 ромбом.

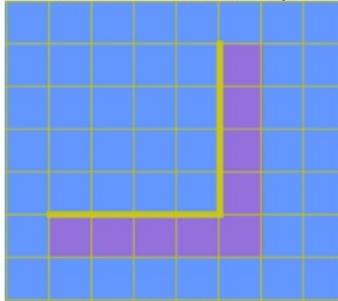


Рис. 2

Составьте из предложенных ниже фрагментов программу, закрашивающую все клетки, расположенные непосредственно правее вертикальной стены, ниже горизонтальной стены и угловую клетку. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка (рис. 1) Робот должен закрасить следующие клетки (см. рис. 2).

Конечное положение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. Алгоритм

должен содержать не более 2-х циклов и вне циклов не более двух подряд идущих команд: вверх, вниз, влево, вправо, закрасить. При исполнении алгоритма Робот не должен разрушиться. Выполнение алгоритма должно завершиться.

*Щелкните лево кнопкой мыши на фрагмент программы, чтобы добавить его в текст программы. Чтобы удалить команду нажмите на крестик напротив этой команды в тексте программы.*

- нц пока слева свободно  нц пока не слева свободно
- нц пока справа свободно  нц пока не справа свободно
- нц пока сверху свободно  нц пока не сверху свободно
- нц пока снизу свободно  нц пока не снизу свободно
- влево  вправо  вверх  вниз  закрасить  кц

- 1 использовать Робот
- 2 алг
- 3 нач
- 4 . нц пока не слева свободно x
- 5 . . закрасить x
- 6 . . вниз x
- 7 . кц x
- 8 . закрасить x
- 9 . влево x
- 10 . нц пока не сверху свободно x
- 11 . . закрасить x
- 12 . . влево x
- 13 . кц x
- ...
- 14 кон

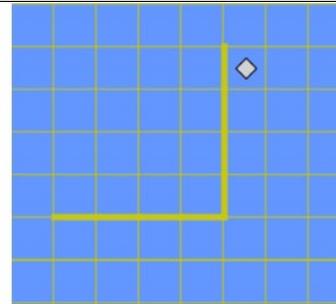


Рис. 1

2

См. критерии

Указания к оцениванию	Баллы
<p>Алгоритм закрашивает все клетки, расположенные непосредственно правее вертикальной стены, ниже горизонтальной стены и угловую клетку. Алгоритм должен содержать 2 цикла и не более двух подряд идущих команд: вниз, влево, вправо, закрасить вне цикла:</p> <pre> 1 использовать Робот 2 алг 3 нач 4 . нц пока не слева свободно      x 5 . . закрасить                    x 6 . . вниз                          x 7 . кц                              x 8 . закрасить                      x 9 . влево                          x 10 . нц пока не сверху свободно     x 11 . . закрасить                    x 12 . . влево                        x 13 . кц                             x ... 14 кон </pre>	2
<p>Алгоритм закрашивает все клетки, расположенные непосредственно правее вертикальной стены, ниже горизонтальной стены, но не закрашивает угловую клетку:</p> <pre> 1 использовать Робот 2 алг 3 нач 4 . нц пока не слева свободно      x 5 . . закрасить                    x 6 . . вниз                          x 7 . кц                              x 8 . влево                          x 9 . нц пока не сверху свободно     x 10 . . закрасить                    x 11 . . влево                        x 12 . кц                             x ... 13 кон </pre> <p>ИЛИ</p> <p>Алгоритм закрашивает все клетки, расположенные непосредственно правее вертикальной стены, закрашивает угловую клетку, но не закрашивает клетки ниже горизонтальной стены:</p>	1

Указания к оцениванию	Баллы
<pre> 1 использовать Робот 2 алг 3 нач 4 . нц пока не слева свободно      x 5 . . закрасить                    x 6 . . вниз                          x 7 . кц                              x 8 . закрасить                      x 9 . влево                          x ... 10 кон </pre> <p>ИЛИ</p> <p>Алгоритм закрашивает угловую клетку, закрашивает клетки ниже горизонтальной стены, но не закрашивает все клетки, расположенные непосредственно правее вертикальной стены:</p> <pre> 1 использовать Робот 2 алг 3 нач 4 . нц пока не слева свободно      x 5 . . вниз                          x 6 . кц                              x 7 . закрасить                      x 8 . влево                          x 9 . нц пока не сверху свободно     x 10 . . закрасить                    x 11 . . влево                        x 12 . кц                             x ... 13 кон </pre>	
Другие варианты	0
<i>Максимальный балл</i>	2