

T-11

11. В одном из произведений А.И. Куприна, текст которого приведён в подкаталоге **Куприн** каталога **PART-2**, присутствует эпизод, в котором Пьер вернулся в цирк с новым номером под другим именем. С помощью поисковых средств операционной системы и текстового редактора выясните, какое имя взял себе Пьер.

T-12

11. В одном из произведений А.С. Грина, текст которого приведён в подкаталоге **Грин** каталога **PART-2**, присутствует повествование о том, как Егер вынул из кармана письмо. С помощью поисковых средств операционной системы и текстового редактора выясните имя человека, от которого было это письмо.

T-13

11. В одном из произведений А.С. Грина, текст которого приведён в подкаталоге **Грин** каталога **PART-2**, присутствует повествование о том, как Аммон увидел четырёх гигантских коров. С помощью поисковых средств операционной системы и текстового редактора выясните кличку коровы цвета жёлтого мыла, с рогами полумесяцем.

T-14

11. В одном из произведений А.С. Грина, текст которого приведён в подкаталоге **Грин** каталога **PART-2**, присутствует повествование о том, как Грей покупал шёлк в лавке у торговца. С помощью поисковых средств операционной системы и текстового редактора выясните, сколько метров шёлка купил Грей.

T-15

11. В одном из произведений Л.Н. Толстого, текст которого приведён в подкаталоге **Толстой** каталога **PART-2**, присутствует повествование о том, что главная героиня Лиза родила девочку. С помощью поисковых средств операционной системы и текстового редактора выясните, в каком месяце это произошло.

T-11

12. Сколько файлов с расширением **.pdf** и **.rtf** содержится в подкаталогах каталогов **Поэзия**? В ответе укажите только число.

T-12

12. Сколько файлов с расширением **.pdf** и **.htm** содержится в подкаталогах каталогов **Поэзия**? В ответе укажите только число.

T-13

12. Сколько файлов с расширением **.rtf** и **.htm** содержится в подкаталогах каталогов **Поэзия** и **Чехов**? В ответе укажите только число.

T-14

12. Сколько файлов с расширением **.txt** и **.doc** содержится в подкаталогах каталогов **Поэзия** и **Чехов**? В ответе укажите только число.

T-15

12. Сколько файлов с расширением **.htm** и **.docx** содержится в подкаталогах каталогов **Толстой** и **Гоголь** каталога **Проза**? В ответе укажите только число.

T-11

15.1 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки.

У Робота есть четыре команды перемещения: **вверх; вниз; влево; вправо.**

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо → соответственно.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения сквозь стену, то он разрушится.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

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**

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**

**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**

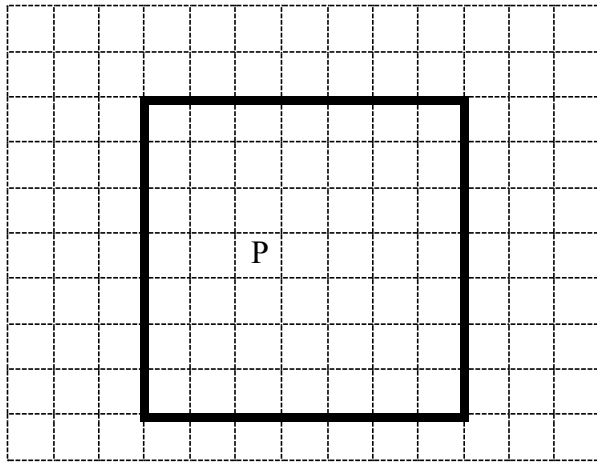
**вправо**

**кц**

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

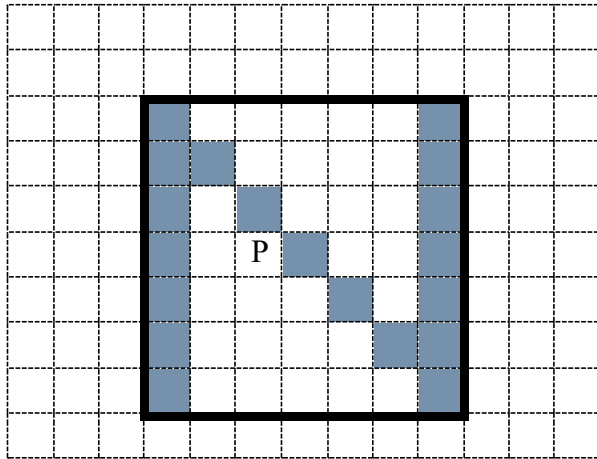
**Выполните задание.**

Робот находится внутри ограниченного стенками **квадрата**. Длины стен **неизвестны**. Одно из возможных начальных расположений Робота приведено на рисунке 1 (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий клетки, расположенные вдоль вертикальных стен, и клетки, расположенные на диагонали, начиная с верхней левой клетки и заканчивая нижней правой клеткой. Робот должен закрасить только клетки, удовлетворяющих данному условию.

Например, для приведённого выше рисунка Робот должен закрасить клетки, заштрихованные на рисунке 2. Конечное расположение робота может быть произвольным. Алгоритм должен решать задачу для бесконечного поля и любого допустимого расположения стен. При исполнении алгоритма Робот не должен разрушиться. Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле.



15.1 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки.

У Робота есть четыре команды перемещения: **вверх; вниз; влево; вправо**.

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо → соответственно.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения сквозь стену, то он разрушится.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

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**

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**

**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**

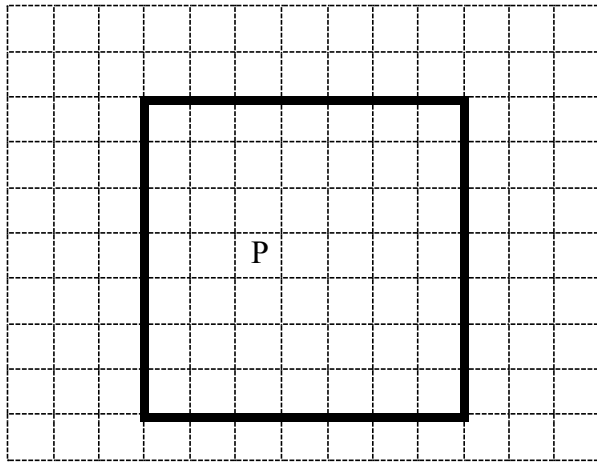
**вправо**

**кц**

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

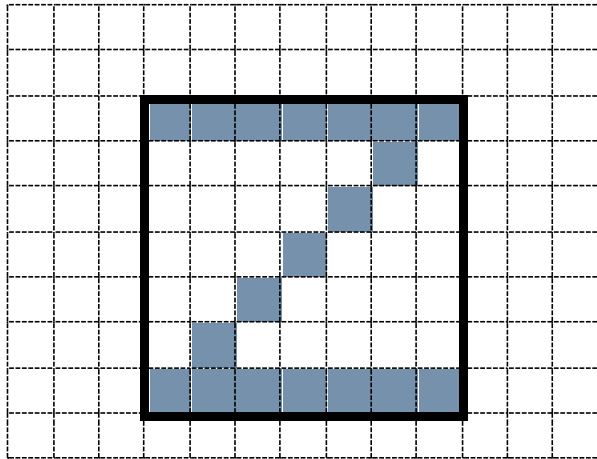
### **Выполните задание.**

Робот находится внутри ограниченного стенками **квадрата**. Длины стен **неизвестны**. Одно из возможных начальных расположений Робота приведено на рисунке 1 (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий клетки, расположенные вдоль вертикальных стен, и клетки, расположенные на диагонали, начиная с верхней левой клетки и заканчивая нижней правой клеткой. Робот должен закрасить только клетки, удовлетворяющих данному условию.

Например, для приведённого выше рисунка Робот должен закрасить клетки, заштрихованные на рисунке 2. Конечное расположение робота может быть произвольным. Алгоритм должен решать задачу для бесконечного поля и любого допустимого расположения стен. При исполнении алгоритма Робот не должен разрушиться. Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле.



15.1 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки.

У Робота есть четыре команды перемещения: **вверх; вниз; влево; вправо**.

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо → соответственно.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения сквозь стену, то он разрушится.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

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**

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**

**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**

Возможно выполнение циклического алгоритма с помощью команды

**нц n раз**

**последовательность команд**

**кц**

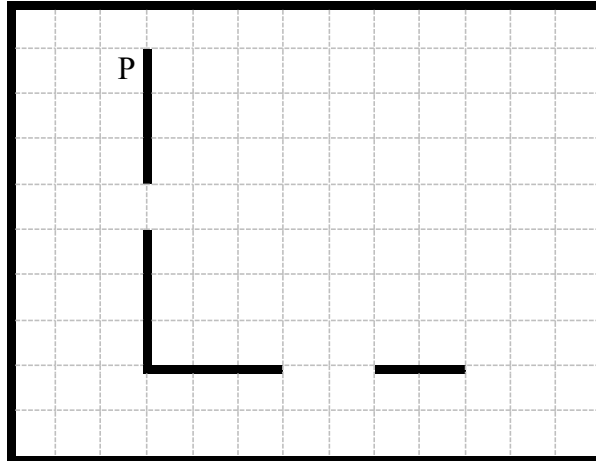
Здесь n – количество повторений.

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

**Выполните задание.**

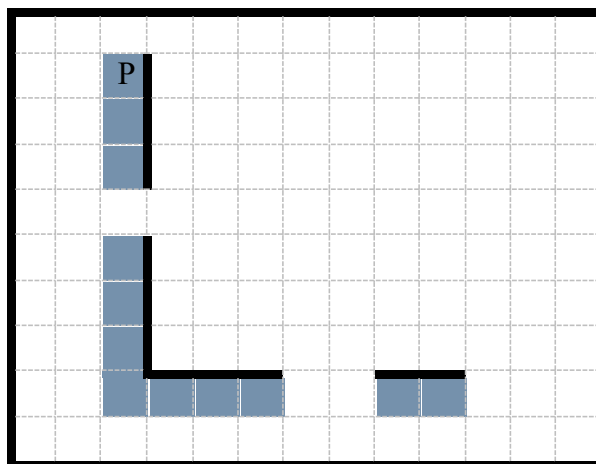
Робот находится внутри ограниченного стенками прямоугольника. Длины стен неизвестны. Внутри прямоугольника находятся горизонтальная и вертикальная стены лабиринта. Левый конец горизонтальной среды лабиринта соединён с нижним концом вертикальной стены. Длины стен лабиринта неизвестны. В каждой стене лабиринта есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной рядом с вертикальной стеной слева от её верхнего конца.

На рисунке 1 находится один из способов расположения стен и Робота (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно левее вертикальной среды и ниже горизонтальной среды. Проходы должны остаться не закрашенными. Требуется закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить клетки, заштрихованные на рисунке 2.

Конечное расположение робота может быть произвольным.



Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При использовании алгоритма Робот не должен разрушиться.

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

Сохраните алгоритм в текстовом файле.



15.1 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки.

У Робота есть четыре команды перемещения: **вверх; вниз; влево; вправо.**

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо → соответственно.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения сквозь стену, то он разрушится.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

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**

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**

**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**

Возможно выполнение циклического алгоритма с помощью команды

**нц n раз**

**последовательность команд**

**кц**

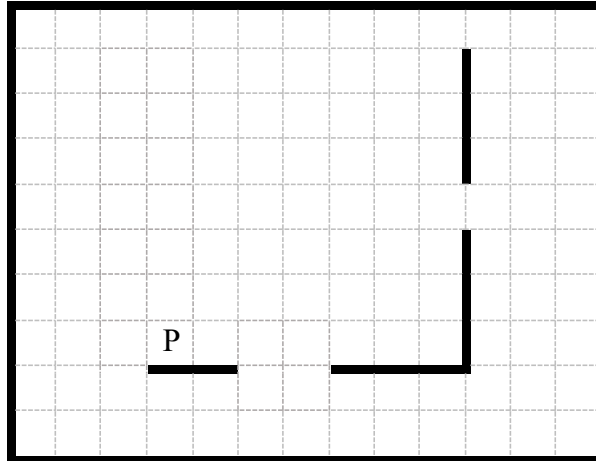
Здесь n – количество повторений.

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

**Выполните задание.**

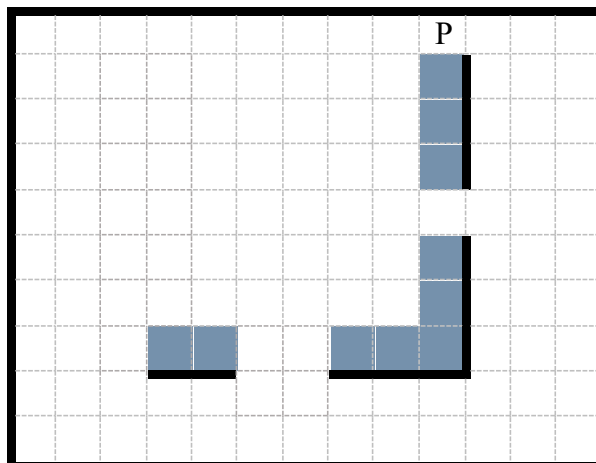
Робот находится внутри ограниченного стенками прямоугольника. Длины стен неизвестны. Внутри прямоугольника находятся горизонтальная и вертикальная стены лабиринта. Левый конец горизонтальной среды лабиринта соединён с нижним концом вертикальной стены. Длины стен лабиринта неизвестны. В каждой стене лабиринта есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной рядом с вертикальной стеной слева от её верхнего конца.

На рисунке 1 находится один из способов расположения стен и Робота (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно левее вертикальной среды и ниже горизонтальной среды. Проходы должны остаться не закрашенными. Требуется закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить клетки, заштрихованные на рисунке 2.

Конечное расположение робота может быть произвольным.



Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При использовании алгоритма Робот не должен разрушиться.

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

Сохраните алгоритм в текстовом файле.

T-15

15.1 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки.

У Робота есть четыре команды перемещения: **вверх; вниз; влево; вправо.**

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо → соответственно.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения сквозь стену, то он разрушится.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

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**

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**

**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**

Возможно выполнение циклического алгоритма с помощью команды

**нц n раз**

**последовательность команд**

**кц**

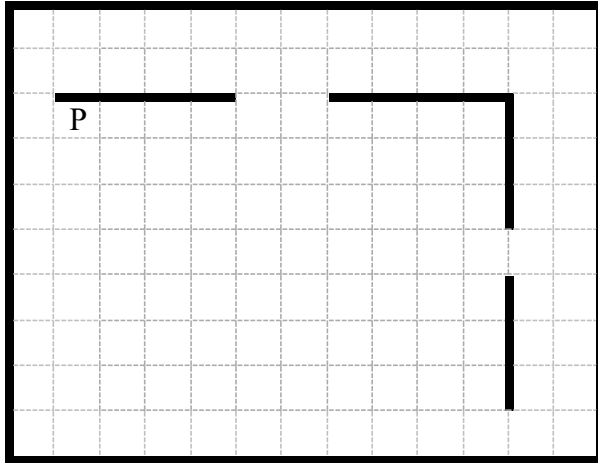
Здесь n – количество повторений.

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

**Выполните задание.**

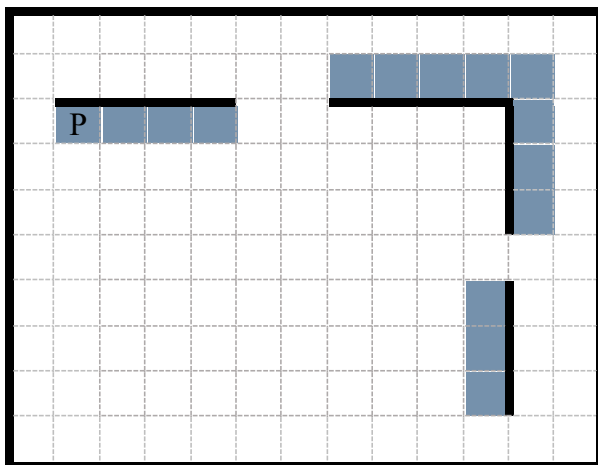
Робот находится внутри ограниченного стенками прямоугольника. Длины стен неизвестны. Внутри прямоугольника находятся горизонтальная и вертикальная стены лабиринта. Левый конец горизонтальной стены лабиринта соединён с нижним концом вертикальной стены. Длины стен лабиринта неизвестны. В каждой стене лабиринта есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной рядом с вертикальной стеной слева от её верхнего конца.

На рисунке 1 находится один из 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 не закрашенными. Требуется закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить клетки, заштрихованные на рисунке 2.

Конечное расположение робота может быть произвольным.



Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При использовании алгоритма Робот не должен разрушиться.

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

Сохраните алгоритм в текстовом файле.