



BIT
EDUCATION
КАДРЫ ДЛЯ ЦИФРОВОЙ ЭКОНОМИКИ



**ВИРТУАЛЬНАЯ
ТВОРЧЕСКАЯ
ЛАБОРАТОРИЯ**

КУРС «ИЗБРАННЫЕ ВОПРОСЫ ПО ИНФОРМАТИКЕ»

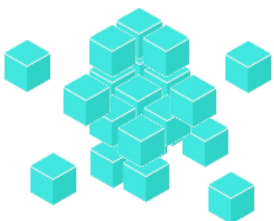
РАЗДЕЛ:

**«АЛГОРИТМИЗАЦИЯ И
ПРОГРАММИРОВАНИЕ»**

ТЕМА ЗАНЯТИЯ:

**РЕШЕНИЕ ЗАДАЧ
ПРАКТИЧЕСКОЙ ЧАСТИ ЭКЗАМЕНА.
РАБОТА С ИСПОЛНИТЕЛЕМ РОБОТ.**

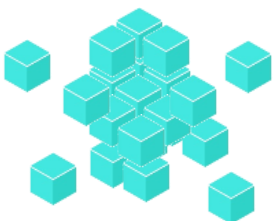




РАСПРЕДЕЛЕНИЕ ЗАДАНИЙ ПО ОСНОВНЫМ ТЕМАТИЧЕСКИМ БЛОКАМ КУРСА ИНФОРМАТИКИ

№ тематического блока	Название тематического блока	№ задания	Какое умение проверяется
2	Алгоритмы и программирование	5	Анализировать простые алгоритмы для конкретного исполнителя с фиксированным набором команд
		6	Формально исполнять алгоритмы, записанные на языке программирования
		15	Создавать и выполнять программы для заданного исполнителя (вариант задания 15.1) или на универсальном языке программирования (вариант задания 15.2)

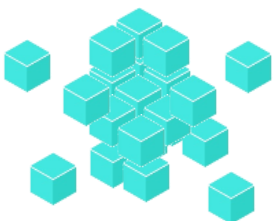




ПЛАН ЭКЗАМЕНАЦИОННОЙ РАБОТЫ

Обозначение уровня сложности задания: Б — базовый, П — повышенный, В — высокий.

Проверяемые элементы содержания и виды деятельности	Уровень сложности задания	Максимальный балл за выполнение задания	Примерное время выполнения задания (мин.)
Задание 6. Формально исполнять алгоритмы, записанные на языке программирования	Б	1	4
Задание 7. Знать принципы адресации в сети Интернет	Б	1	3
Задание 8. Понимать принципы поиска информации в Интернете	П	1	5
Задание 9. Умение анализировать информацию, представленную в виде схем	П	1	4
Задание 10. Записывать числа в различных системах счисления	Б	1	3
Задание 11. Поиск информации в файлах и каталогах компьютера	Б	1	6
Задание 12. Определение количества и информационного объёма файлов, отобранных по некоторому условию	Б	1	6
Задание 13. Создавать презентации (вариант задания 13.1) или создавать текстовый документ (вариант задания 13.2)	П	2	25
Задание 14. Умение проводить обработку большого массива данных с использованием средств электронной таблицы	В	3	30
Задание 15. Создавать и выполнять программы для заданного исполнителя (вариант задания 15.1) или на универсальном языке программирования (вариант задания 15.2)	В	2	45



ЗАДАНИЕ №15.1: «СОЗДАВАТЬ И ВЫПОЛНЯТЬ ПРОГРАММЫ ДЛЯ ЗАДАННОГО ИСПОЛНИТЕЛЯ»

УРОВЕНЬ СЛОЖНОСТИ – ВЫСОКИЙ.

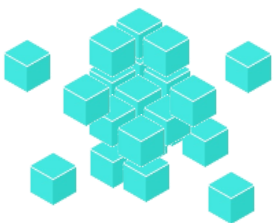
МАКСИМАЛЬНЫЙ БАЛЛ – 2 БАЛЛА.

ПРИМЕРНОЕ ВРЕМЯ ВЫПОЛНЕНИЯ – 45 МИНУТ.

ПРОВЕРЯЕМОЕ УМЕНИЕ:

- ИСПОЛЬЗОВАТЬ СТАНДАРТНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ ДЛЯ ПОСТРОЕНИЯ АЛГОРИТМОВ ДЛЯ ФОРМАЛЬНЫХ ИСПОЛНИТЕЛЕЙ;
- ФОРМАЛЬНО ИСПОЛНЯТЬ АЛГОРИТМЫ, ЗАПИСАННЫЕ НА ЕСТЕСТВЕННОМ И АЛГОРИТМИЧЕСКОМ ЯЗЫКАХ;
- РАЗРАБОТКА АЛГОРИТМА ДЛЯ ФОРМАЛЬНОГО ИСПОЛНИТЕЛЯ ИЛИ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ С ИСПОЛЬЗОВАНИЕМ УСЛОВНЫХ ИНСТРУКЦИЙ И ЦИКЛОВ, А ТАКЖЕ ЛОГИЧЕСКИХ СВЯЗОК ПРИ ЗАДАНИИ УСЛОВИЙ.





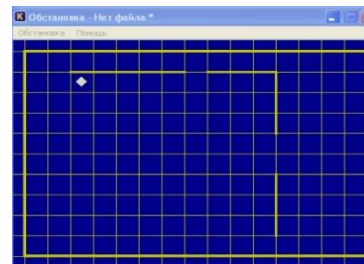
РЕКОМЕНДАЦИИ:



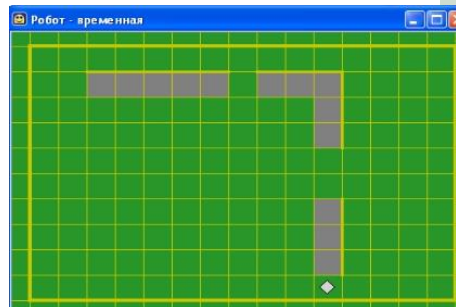
Рекомендации по выполнению

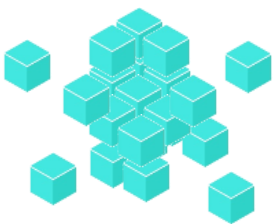
Для выполнения этого задания необходимо:

1. сформулируйте последовательно условия движения Робота вдоль объектов и закрашивания им клеток,
2. затем составьте алгоритм, запишите его в виде программы,
3. проверьте правильность работы программы в среде разработки «КуМир» (или аналогичной) на различных исходных конфигурациях .



```
К Новая программа - КуМир
Программа Редактирование Вставка Выполнение Инструменты Робот Чертежики
1 использовать Робот
2 алг
3 нач
4   • нц пока не сверху свободно
5   •   • закрасить
6   •   • вправо
7   •   • кц
8   •   • нц пока сверху свободно
9   •   •   • вправо
10  •   •   • кц
11  •   •   • нц пока не сверху свободно и справа свободно
12  •   •   •   • закрасить
13  •   •   •   • вправо
14  •   •   •   • кц
15  •   •   •   • нц пока не справа свободно
16  •   •   •   •   • закрасить
17  •   •   •   •   • вниз
18  •   •   •   •   • кц
19  •   •   •   •   • нц пока справа свободно
20  •   •   •   •   •   • вниз
21  •   •   •   •   •   • кц
22  •   •   •   •   •   • нц пока не справа свободно
23  •   •   •   •   •   •   • закрасить
24  •   •   •   •   •   •   • вниз
25  •   •   •   •   •   •   • кц
26  •   •   •   •   •   •   • нц пока справа свободно
27  •   •   •   •   •   •   •   • закрасить
28  •   •   •   •   •   •   •   • вниз
29  •   •   •   •   •   •   •   • кц
30 кон
```





ТИПИЧНЫЕ ОШИБКИ



Рекомендации по их предотвращению

Обратите внимание, в условии сказано:

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

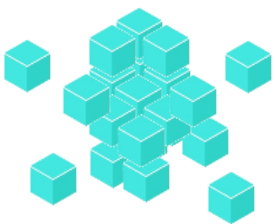
Это означает, что нужно обязательно проверить работу алгоритма на различных допустимых конфигурациях.

То, что программа успешно прошла тестирование, не гарантирует отсутствия в ней ошибок, но существенно снижает их вероятность.

Выбирайте в том числе сложные тесты.

Помните, что цель тестирования – не продемонстрировать самому себе правильность работы программы, а выявить и своевременно устранить возможную ошибку .





ЧТО НУЖНО ПОМНИТЬ:



Цикл «ПОКА»

ИСПОЛЬЗОВАТЬ **Робот**

алг

нач

нц **пока** справа свободно

вправо

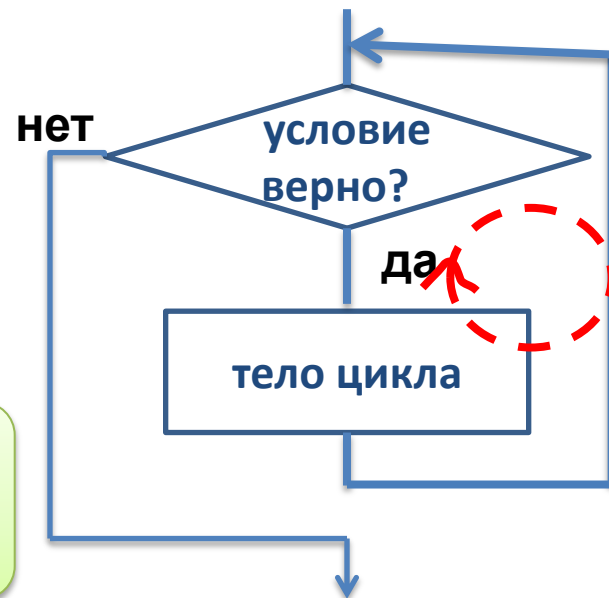
кц

закрасить

кон

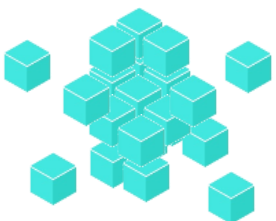
условие повторения

тело цикла – это команды, которые выполняются несколько раз



Цикл с условием – это цикл, который выполняется до тех пор, пока условие не станет ложным.

Количество шагов такого цикла определяется исходными данными.



ЧТО НУЖНО ПОМНИТЬ:



Логические команды

Логическая команда – это вопрос, на который исполнитель отвечает «да» или «нет».

С помощью логических команд (команд-вопросов) Робот может проверять условия, связанные с окружающей обстановкой. Например, для того чтобы определить, можно ли двигаться в каком-то направлении, нужно выяснить, нет ли там стены.

10 команд проверки условий

сверху стена

сверху свободно

справа стена

справа свободно

снизу стена

снизу свободно

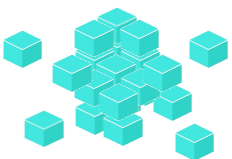
слева стена

слева свободно

клетка закрашена

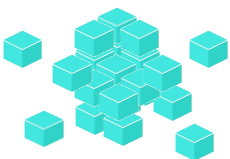
клетка чистая





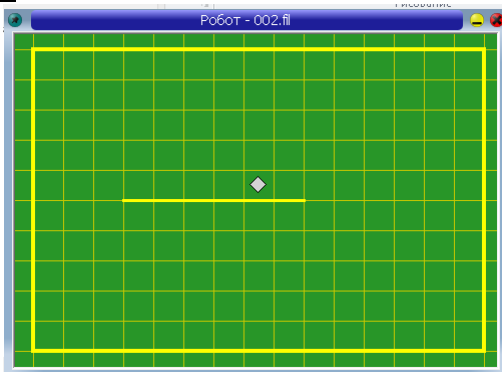
ПРАКТИЧЕСКАЯ ЧАСТЬ:





РАССМОТРИМ ЗАДАНИЕ 15.1

15.1 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

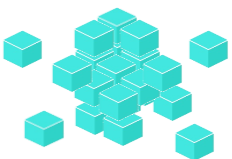
При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо →.

Если Робот получит команду передвижения сквозь стену, то он разрушится.

Также у Робота есть команда **закрасить**, при которой закрашивается клетка, в которой 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**

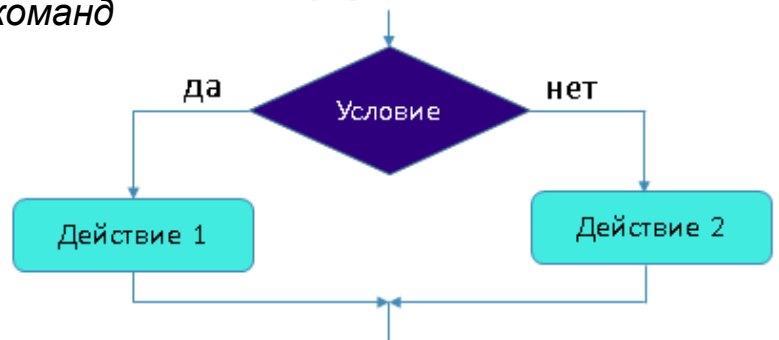


РАССМОТРИМ ЗАДАНИЕ 15.1

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

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

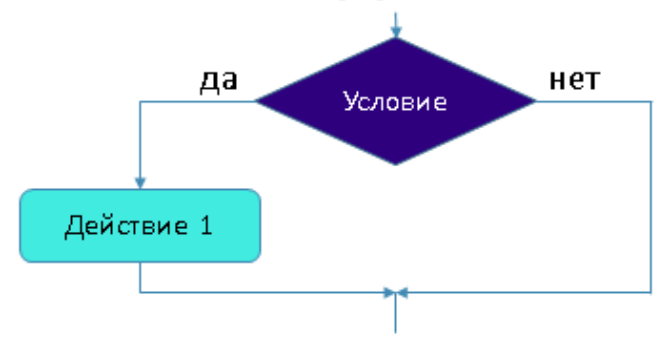
Полная форма ветвления



Формат блока:

```
нач  
  если <Условие>  
    то <Действие_1>  
    иначе <Действие_2>  
все  
кон
```

Неполная форма ветвления



Формат блока:

```
нач  
  если <Условие>  
    то <Действие_1>  
все  
кон
```

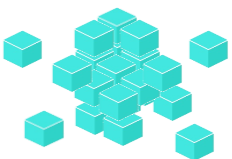
Здесь *условие* — одна из команд проверки условия.

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





[РАССМОТРИМ ЗАДАНИЕ 15.1]

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**

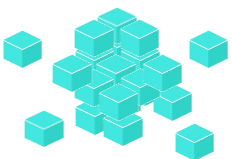
Для повторения последовательности команд можно использовать цикл **«пока»**, имеющий следующий вид:

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

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

**нц пока справа свободно
вправо
кц**





ПРИМЕР №1:



Выполните задание:

На бесконечном поле имеется
длинная вертикальная стена.

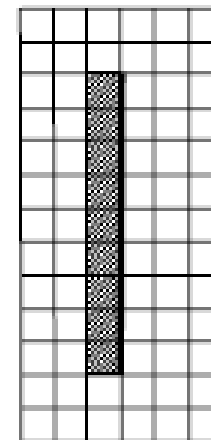
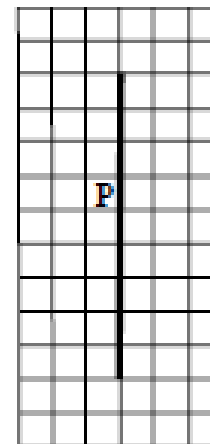
Длина стены неизвестна.

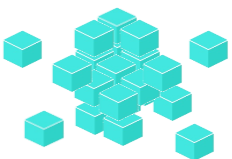
Робот находится в одной из клеток,
расположенной непосредственно слева от стены.

Начальное положение робота также неизвестно. Одно из
возможных положений робота приведено на рисунке (робот
обозначен буквой «Р»).

Напишите для робота алгоритм, закрашивающий все клетки,
расположенные левее стены и прилегающие к ней. Робот
должен закрасить только клетки, удовлетворяющие данному
условию.

Например, для приведенного выше рисунка робот должен
закрасить следующие клетки .





ПРИМЕР №1:

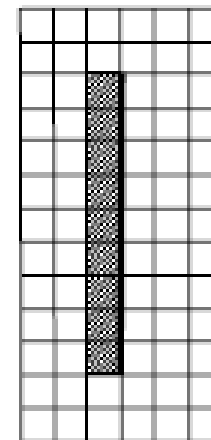
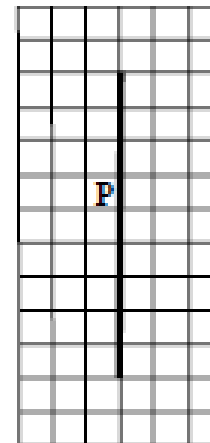


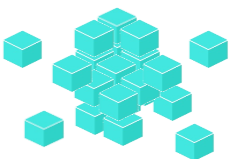
При исполнении алгоритма Робот не должен разрушиться, выполнение алгоритма должно завершиться. Конечное расположение Робота может быть произвольным.

Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен.

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

Сохраните алгоритм в файле. Название файла и каталога для сохранения Вам сообщать организации.





РЕШЕНИЕ ПРИМЕРА №1:

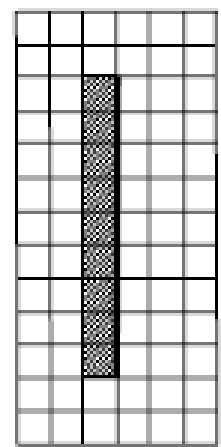
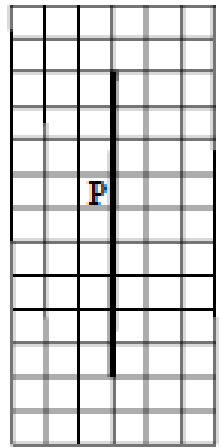
К Новая программа - Кумир

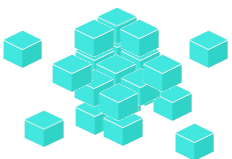
Программа Редактирование Вставка Выполнение Инструменты Робот Чертежник Инфо Миры

использовать **Робот**

алг
нач
▪ нц пока справа стена
▪ ▪ вверх
▪ кц
▪ вниз
▪ нц пока справа стена
▪ ▪ закрасить
▪ ▪ вниз
▪ кц
кон

Робот - 1.fil





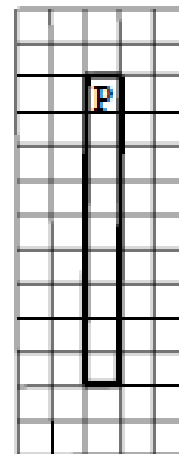
ПРИМЕР №2:



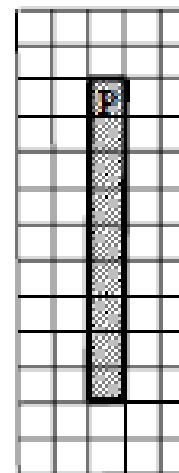
Выполните задание:

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

Возможный вариант начального расположения Робота приведён на рисунке (Робот обозначен буквой «Р»)

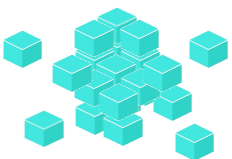


Начальное положение робота также неизвестно. Одно из возможных положений робота приведено на рисунке (робот обозначен буквой «Р»).



Напишите для робота алгоритм, закрашивающий все клетки, расположенные левее стены и прилегающие к ней. Робот должен закрасить только клетки, удовлетворяющие данному условию.

Например, для приведенного выше рисунка робот должен закрасить следующие клетки .



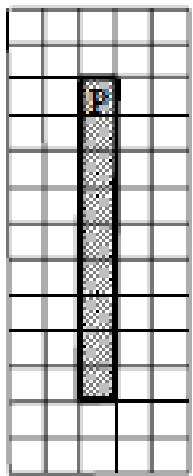
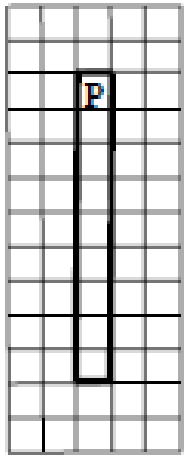
РЕШЕНИЕ ПРИМЕРА №2:

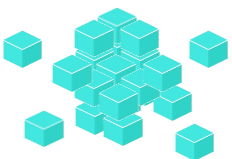
К Новая программа - Кумир

Программа Редактирование Вставка Выполнение Инструменты Робот Чертежник Инфо Миры

использовать Робот

```
алг
нач
▪ нц пока снизу свободно
▪ ▪ вниз
▪ кц
▪ нц пока сверху свободно
▪ ▪ закрасить
▪ ▪ вверх
▪ кц
▪ закрасить
кон
```





ПРИМЕР №3:

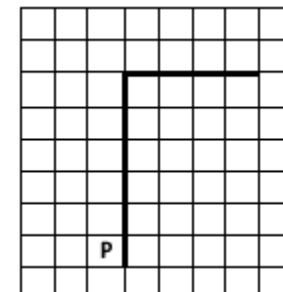


Выполните задание:

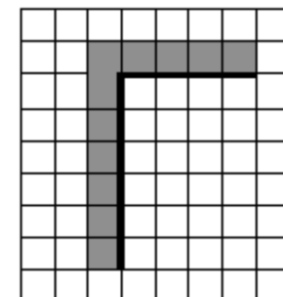
На бесконечном поле имеется вертикальная стена.

Длина стены неизвестна. От верхнего конца стены вправо отходит горизонтальная стена **также неизвестной длины.**

Возможный вариант начального расположения Робота приведён на рисунке (Робот обозначен буквой «Р»)

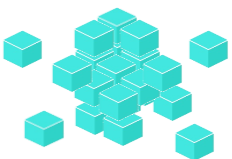


Начальное положение робота также неизвестно. Одно из возможных положений робота приведено на рисунке (робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные левее вертикальной стены и выше горизонтальной стены и прилегающие к ним. Робот должен закрасить только клетки, удовлетворяющие данному условию.

Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок)



РЕШЕНИЕ ПРИМЕРА №3:

К Новая программа - Кумир

Программа Редактирование Вставка Выполнение Инструменты Робот Чертежник Инфо Миры

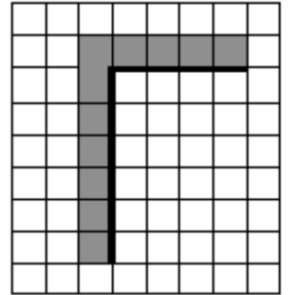
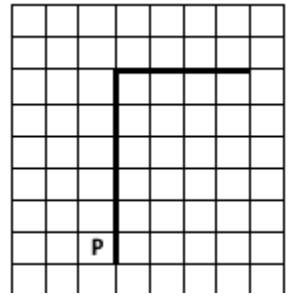
использовать **Робот**

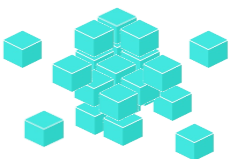
алг
нач

- нц пока справа стена
- ▪ закрасить
- ▪ вверх
- кц
- закрасить
- вправо
- нц пока снизу стена
- ▪ закрасить
- ▪ вправо
- кц

кон

Робот - 3.fil





[САМОСТОЯТЕЛЬНО РЕШИМ:]

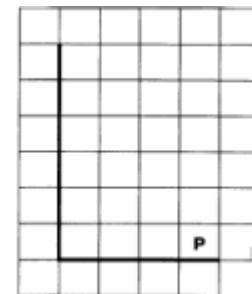
Выполните задание:

На бесконечном поле имеется вертикальная стена.

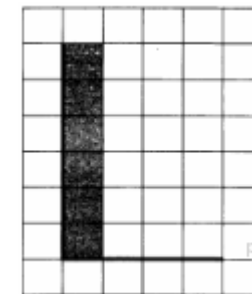
Длина стены неизвестна. От нижнего конца стены вправо отходит горизонтальная стена также **неизвестной длины**.

Робот находится в клетке, расположенной над правым краем горизонтальной стены.

Возможный вариант начального расположения Робота приведён на рисунке (Робот обозначен буквой «Р»)



Начальное положение робота также неизвестно. Одно из возможных положений робота приведено на рисунке (робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные левее вертикальной стены и выше горизонтальной стены и прилегающие к ним. Робот должен закрасить только клетки, удовлетворяющие данному условию.

Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок)