

Исполнитель Робот



Применение системы «Исполнители»
(К.Поляков, <http://kpolyakov.narod.ru>)
при изучении раздела «Алгоритмика»
пропедевтического курса Информатики и ИКТ

Содержание

1. Исполнитель Робот. Управление Роботом
2. Исполнитель Робот. Цикл «Пока»
3. Исполнитель Робот. Ветвление

1. Исполнитель Робот Управление Роботом



Давайте вспомним

- Исполнитель – это ...
- Управление – это ...
- Алгоритм – это ...
- Опишите исполнителя Чертёжника по плану:
 - 1) Имя
 - 2) Круг решаемых задач
 - 3) Среда
 - 4) СКИ
 - 5) Система отказов
 - 6) Режимы работы



Система «Исполнители»

The screenshot displays the 'Система "Исполнители"' (System "Executors") interface. The window title is 'Система "Исполнители"'. The menu bar includes 'Файл', 'Правка', 'Настройка', 'Шаблоны', and 'Лабиринт'. The toolbar contains various icons for file operations and execution. The main area is split into two panes:

- Left Pane:** A code editor titled 'Программа' (Program) containing a simple C-style block structure:

```
{  
 |  
}
```
- Right Pane:** A maze task titled 'Задача для Робота: z1.maz'. The maze is a 10x10 grid with a robot at the bottom center, a goal 'P' at the top right, and a wall in the center. A callout 'Восстановить лабиринт' (Restore the maze) points to the maze grid.

Yellow callout boxes provide additional labels:

- 'Режим Робота' (Robot Mode) is located at the top center.
- 'Загрузить задачу для Робота' (Load task for the Robot) is at the top right.
- 'Окно редактора программ' (Code editor window) is on the left side.
- 'Задача для Робота' (Robot task) is on the right side.
- 'Поле Исполнителя' (Executor field) is at the bottom center.

The status bar at the bottom left shows '3:3'. The bottom right corner features a Windows 2.5 logo and navigation arrows.

Среда Робота

Робот выполняет специальную задачу - сажает цветы в грядки на поле (лабиринте).

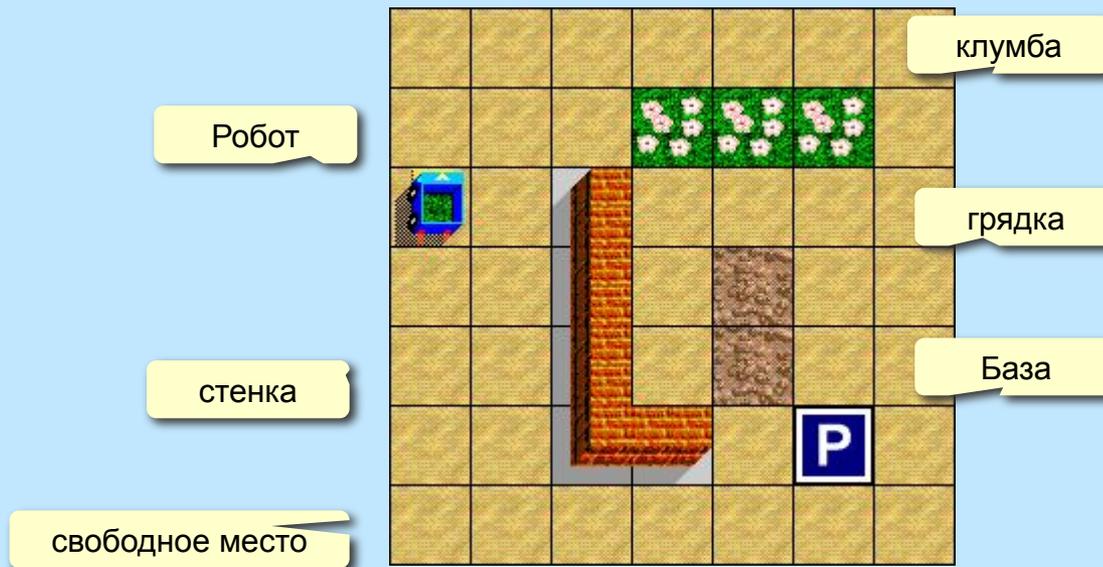
Поле размечено на квадраты, каждый из которых может быть:

- 1) свободным местом;
- 2) грядкой;
- 3) клумбой с цветами;
- 3) стенкой .

Робот может переходить из клетки в клетку по грядкам или по свободным клеткам.

Робот не может проходить через стенки, ходить по клумбам с цветами, выходить за границы поля.

Робот должен посадить цветы на всех грядках и вернуться на Базу для пополнения запасов.



СКИ Работа

Основные команды:

направо; - повернуться на 90 градусов вправо

налево; - повернуться на 90 градусов влево

кругом; - развернуться кругом (на 180 градусов)

вперед (n); - перейти на n клеток вперед

назад (n); - перейти на n клеток назад

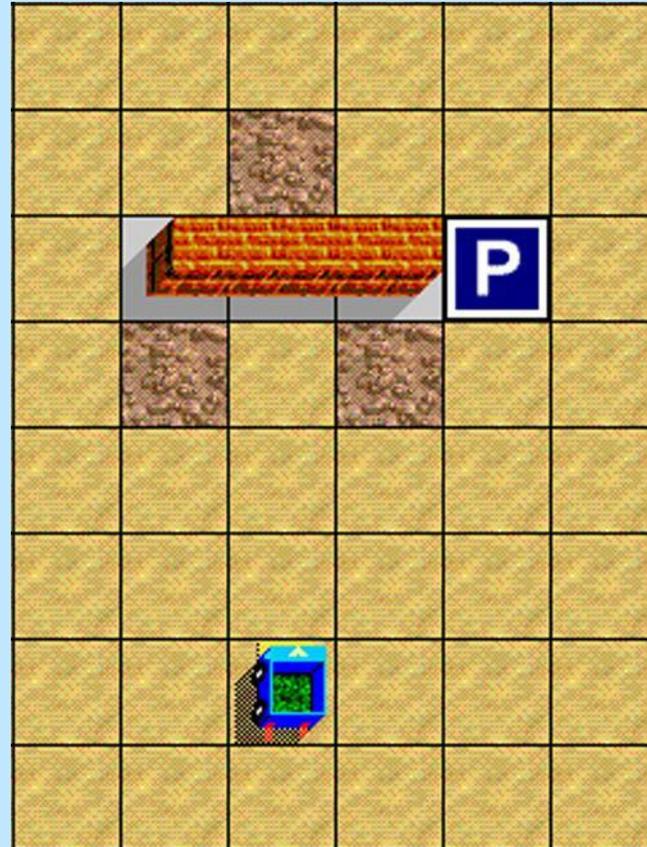
посади; - посадить цветы на грядке, где стоит Робот



Пример простой задачи

Задача1

```
{  
  вперед ( 3 );  
  налево;  
  назад ( 1 );  
  посади;  
  вперед ( 2 );  
  посади;  
  вперед ( 1 );  
  направо;  
  вперед ( 2 );  
  направо;  
  вперед ( 2 );  
  посади;  
  вперед ( 2 );  
  направо;  
  вперед ( 1 );  
}
```



Возможные ошибки Робота

- 1. Синтаксические (“НЕ ПОНИМАЮ”)** – появляются при ошибках в написании команд, например:
влево;
вперет (3);
направо (2);
- 2. Отказы (“НЕ МОГУ”)** – появляются, например, если Роботу дают команду идти прямо на стенку или сажать цветы там, где нет грядки.
- 3. Логические** – возникают тогда, когда Робот понимает команды и выполняет их, но результат не тот, какой нужен.



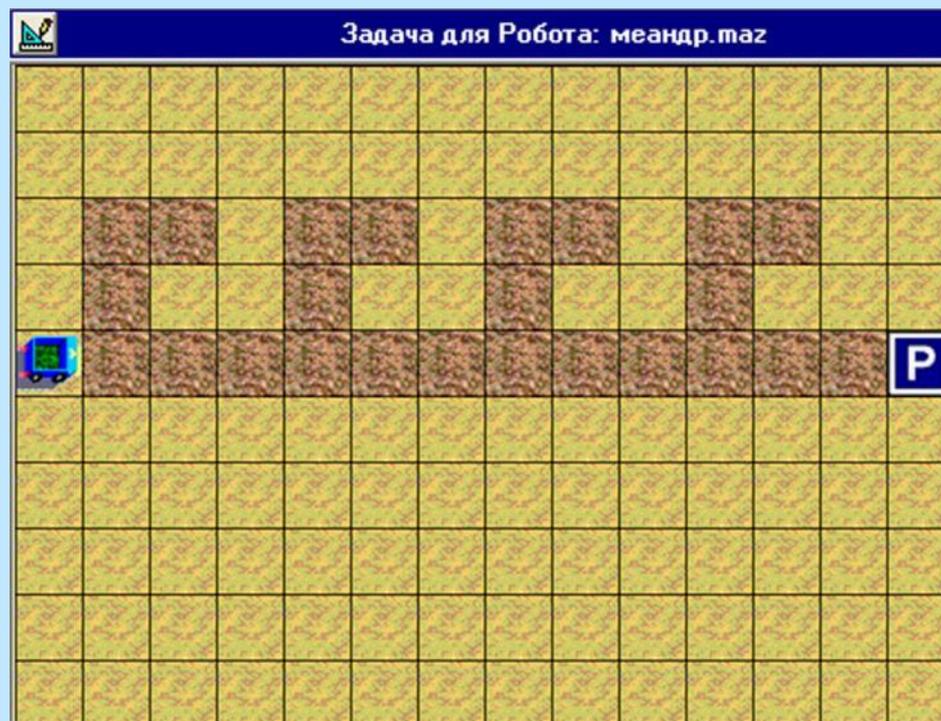
Пример алгоритма управления Роботом

Составить программу, после выполнения которой Робот посадит цветы в рядку в форме меандра из 4 витков и придет на Базу.

1 способ

Меандр1

```
{
  вперед ( 1 );
  повтори ( 4 )
  {
    налево;
    посади; вперед ( 1 );
    посади; вперед ( 1 );
    посади;
    направо; вперед ( 1 );
    посади; направо;
    вперед ( 2 ); налево;
    посади; вперед ( 1 );
    посади; вперед ( 1 );
  }
}
```



Пример алгоритма управления Роботом

Составить программу, после выполнения которой Робот посадит цветы в рядку в форме меандра из 4 витков и придет на Базу.

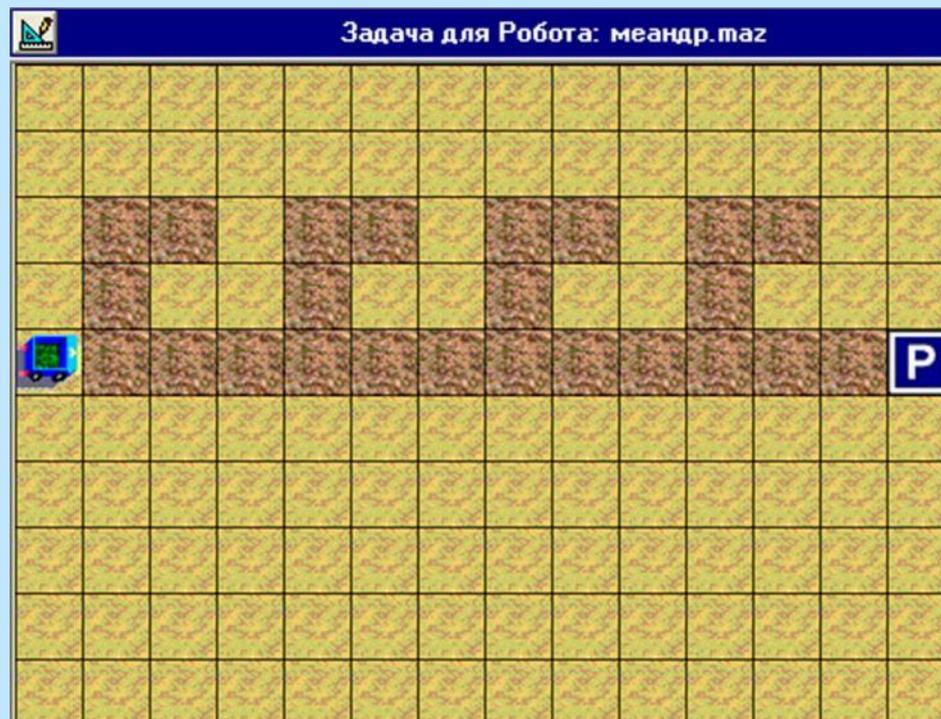
2 способ
(с использованием процедуры)

Меандр2

```
{  
  вперед ( 1 );  
  повтори ( 4 ) Виток ;  
}
```

Виток

```
{  
  налево ;  
  посади ; вперед ( 1 );  
  посади ; вперед ( 1 );  
  посади ;  
  направо ; вперед ( 1 );  
  посади ; направо ;  
  вперед ( 2 ) ; налево ;  
  посади ; вперед ( 1 );  
  посади ; вперед ( 1 );  
}
```



Задача 1

(РТ №22 стр. 99, учебник №1 стр.148)

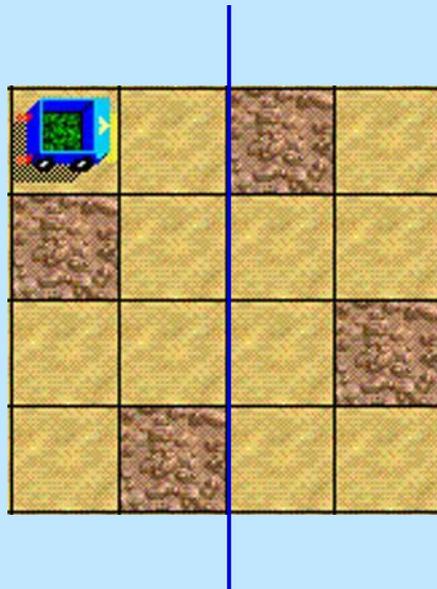
Приведите все алгоритмы из трех команд, которые переместят Робота из исходного положения на Базу.



Задача 2

(РТ №23 стр. 99, учебник №4 стр.148)

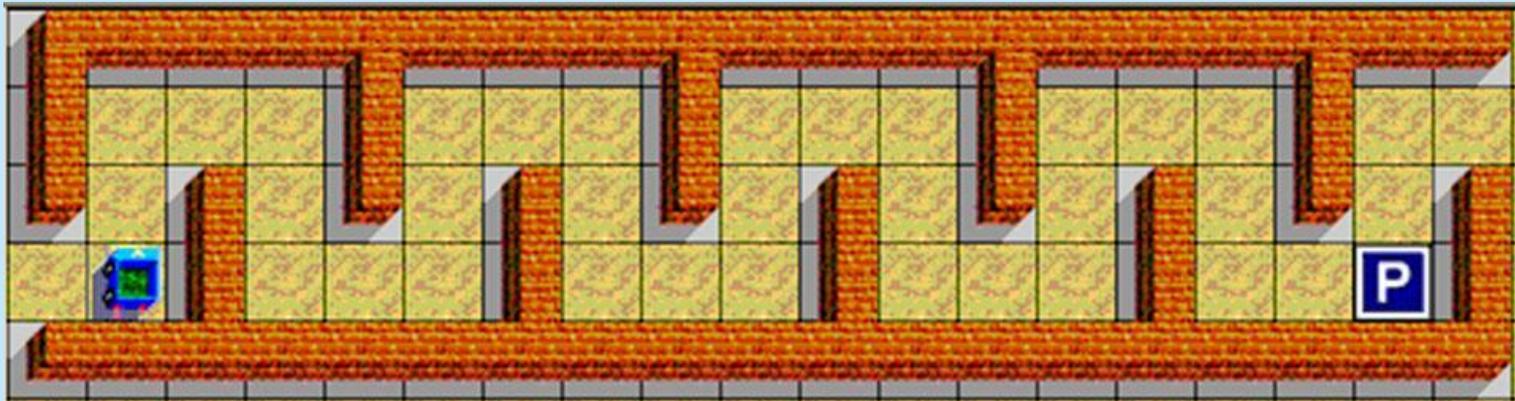
Маша придумала лабиринт для Робота. Коля стёр ровно половину клеток с грядками. Восстановите рисунок (он симметричен относительно вертикальной оси). Напишите программу для Робота.



Задача 3

(РТ №25 стр. 100, учебник №6 стр.149)

Напишите программу, с помощью которой Робот пройдет по лабиринту и попадет на Базу .



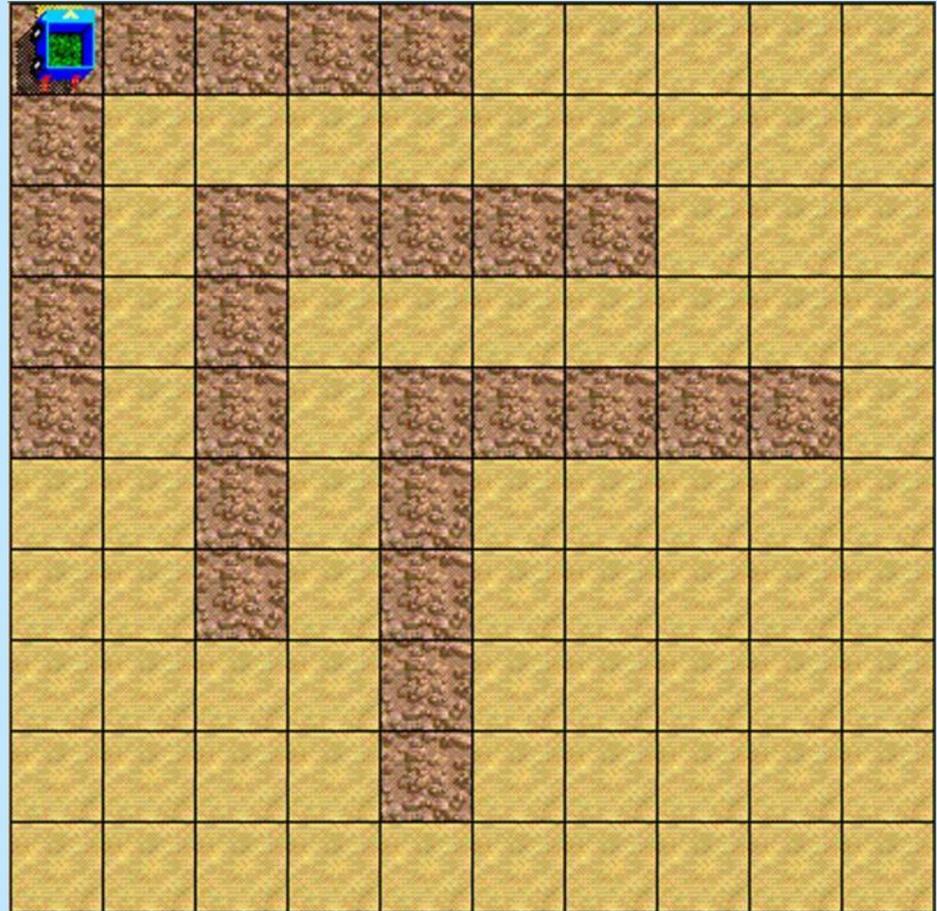
Практическая работа

(РТ №27(а) стр. 103, учебник №8(а) стр.149)

Напишите программу, с помощью которой Робот посадит цветы на грядках в соответствии с рисунком .

РТ27а

```
{
повтори ( 3 )
{
Угол;
направо; вперед ( 2 );
направо; вперед ( 2 );
направо;
}
}
Угол
{
назад ( 4 );
посади; вперед ( 1 );
направо;
посади; вперед ( 1 );
посади;
}
```



Домашнее задание

§3.3 стр. 135-136

РТ №24 стр. 100, учебник №5 стр. 148

Напишите программу, с помощью которой Робот сможет попасть на Базу во всех трех лабиринтах .



а)



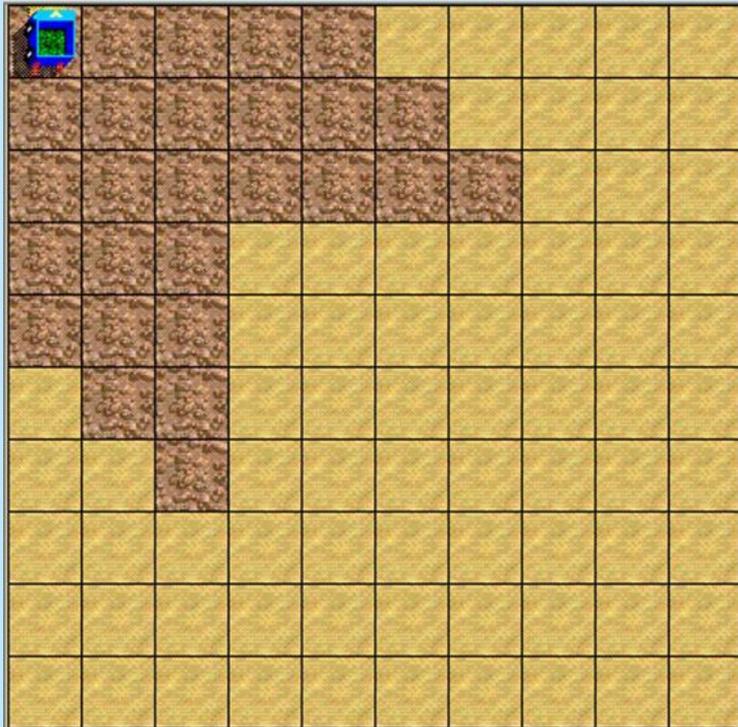
б)



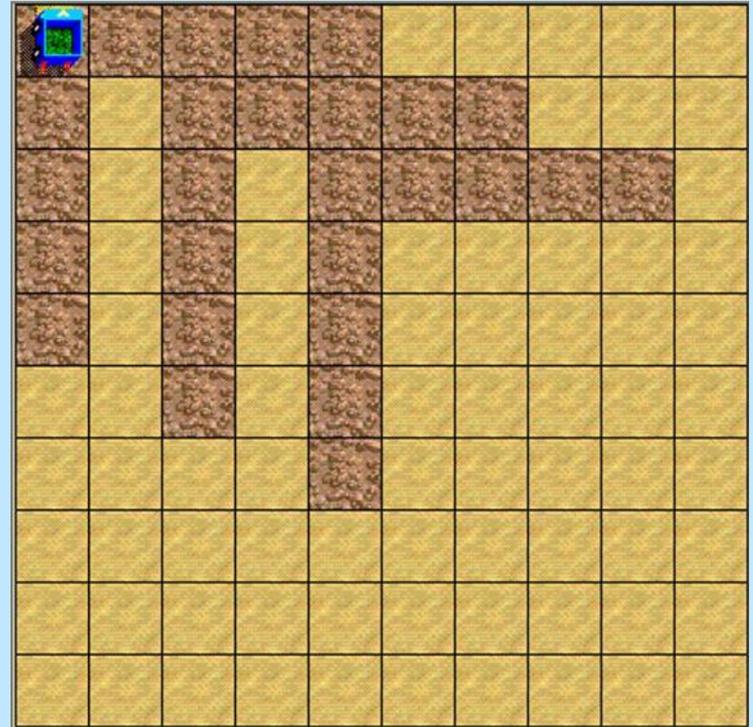
в)

Дополнительное домашнее задание

РТ №27 стр. 103, учебник №8 стр. 149



б)



в)

2. Исполнитель Робот Цикл «Пока»



Давайте вспомним

- Неформальный исполнитель – это ...
- Приведите примеры неформальных исполнителей ...
- Формальный исполнитель – это ...
- Приведите примеры формальных исполнителей
- Опишите исполнителя Работа по плану:
 - 1) Имя
 - 2) Круг решаемых задач
 - 3) Среда
 - 4) СКИ
 - 5) Система отказов
 - 6) Режимы работы



Алгоритмы с обратной связью

Робот может получать информацию об окружающей обстановке с помощью датчиков, которые выдают ответ "Да" или "Нет" ("истинно" или "ложно") на вопросы-команды.

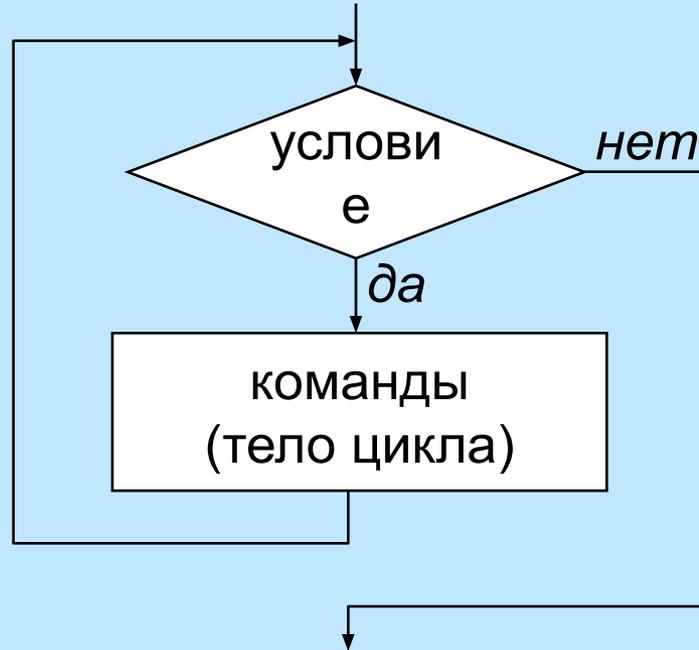
Робот может проверять следующие **простые условия**:

слева_стена справа_стена впереди_стена сзади_стена	эти команды определяют, есть ли стена в ближайшей клетке в указанном направлении
слева_клумба справа_клумба впереди_клумба сзади_клумба	эти команды определяют, есть ли клумба в соседней клетке в указанном направлении
слева_свободно справа_свободно впереди_свободно сзади_свободно	эти команды определяют, свободна ли ближайшая клетка в указанном направлении (Робот получает ответ "Да", если там нет стены и нет клумбы)
рядка клумба база	эти команды определяют, является ли клетка, в которой стоит Робот, рядкой клумбой или Базой



Цикл «Пока»

```
пока ( условие )  
{  
  команды ;  
}
```

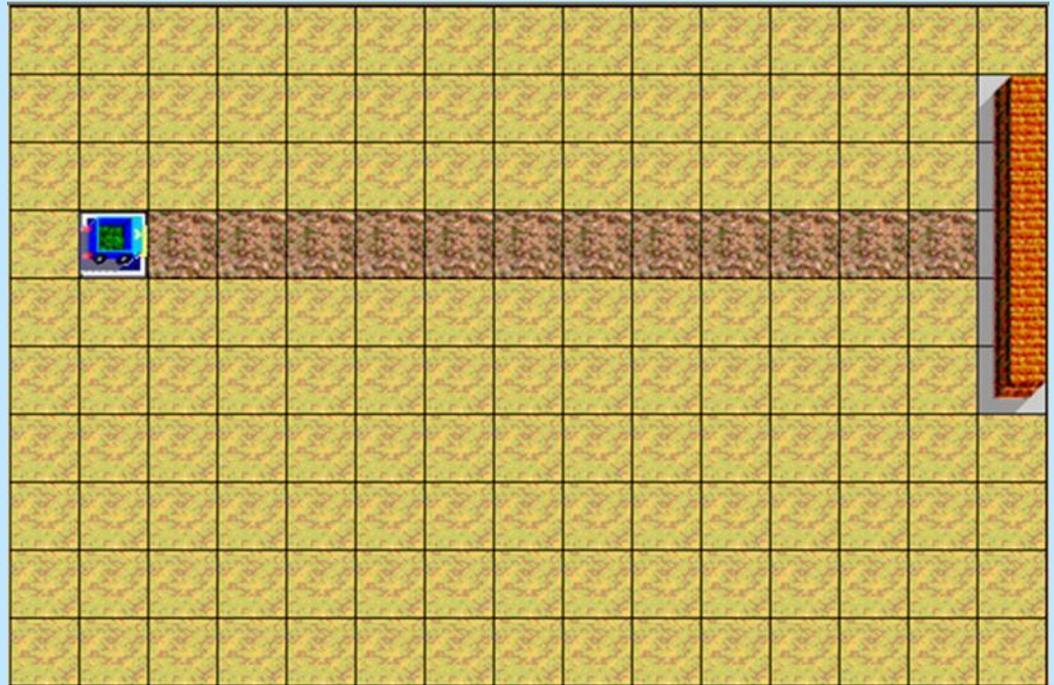


Пример задачи с циклом «пока»

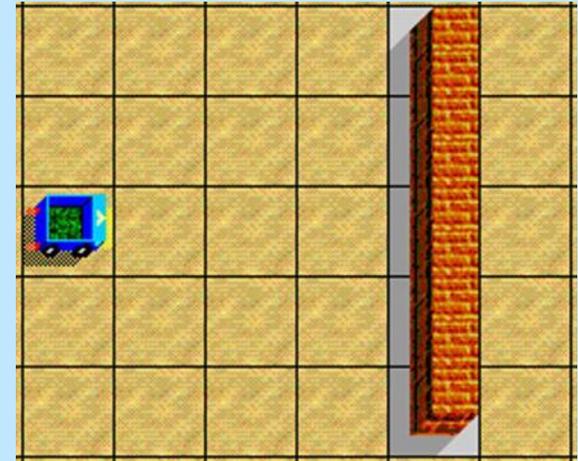
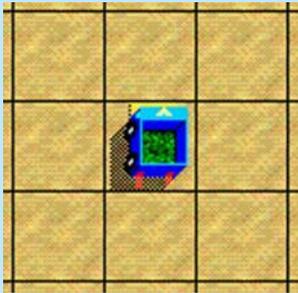
Составить программу, после выполнения которой Робот посадит цветы в прямолинейную грядку неизвестной длины до стены и вернется на Базу.

Длинная_клуба

```
{
пока ( впереди_свободно )
{
вперед ( 1 );
посади;
}
налево;
вперед ( 1 );
налево;
пока ( слева_клуба )
{
вперед ( 1 );
}
налево; вперед ( 1 );
}
```



Примеры ошибок в циклах «пока»



Ничего

```
{  
  пока ( справа_стена )  
  {  
    вперед ( 1 );  
  }  
}
```

Защикливание

```
{  
  пока ( справа_стена )  
  {  
    кругом; кругом;  
  }  
}
```

Не_могу

```
{  
  пока ( вперед_свободно )  
  {  
    вперед ( 2 );  
  }  
}
```

Составные условия

Составные условия образуются из простых условий добавлением логических операций **И**, **ИЛИ**, **НЕ**.

Пусть **A**, **B** – простые условия.

Составное условие (**A и B**) будет выполняться только тогда, когда выполняются каждое из простых условий.

Составное условие (**A или B**) будет выполняться тогда, когда выполняется хотя бы одно из простых условий.

Составное условие (**не A**) будет выполняться, когда не выполняется простое условие **A**.

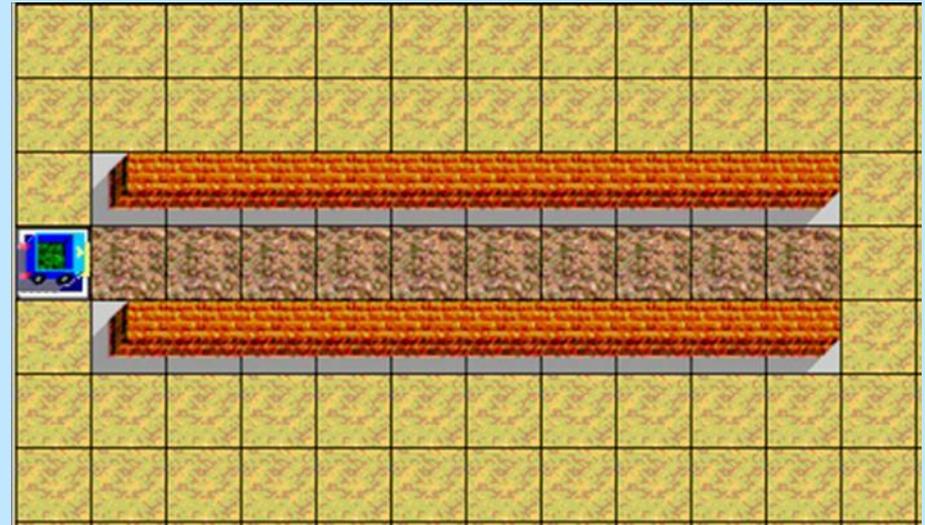
Практическая работа

Пример 1

Составить программу, после выполнения которой Робот посадит цветы в прямом коридоре неизвестной длины и вернется на Базу.

Клумба_в_коридоре

```
{  
вперед ( 1 );  
пока ( слева_стена и справа_стена )  
{  
посади; вперед ( 1 );  
}  
налево; вперед ( 2 );  
налево; вперед ( 1 );  
пока ( слева_стена )  
{  
вперед ( 1 );  
}  
налево; вперед ( 2 );  
}
```



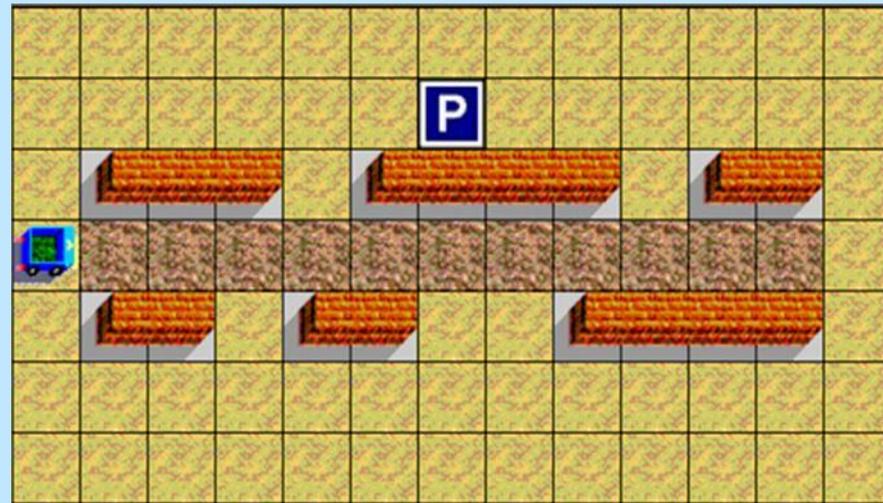
Практическая работа

Пример 2

Составить программу, после выполнения которой Робот посадит цветы в прямом коридоре с проходами неизвестной длины и вернется на Базу.

Клумба_в_коридоре2

```
{  
вперед ( 1 );  
пока ( слева_стена или справа_стена )  
{  
посади; вперед ( 1 );  
}  
налево; вперед ( 2 );  
налево; вперед ( 1 );  
пока ( не база ) вперед ( 1 );  
}
```

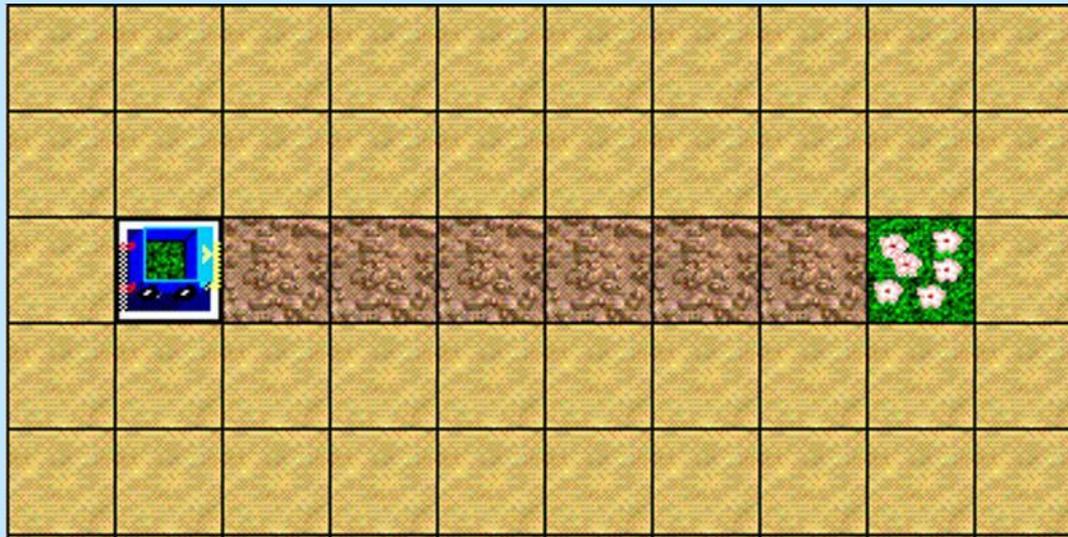


Домашнее задание

§3.3 стр. 137-144

РТ №30 стр. 105, учебник №10 стр. 150

Напишите программу, с помощью которой Робот посадит цветы в грядке до уже посаженной клумбы и вернется в исходное положение.



3. Исполнитель Робот Ветвление



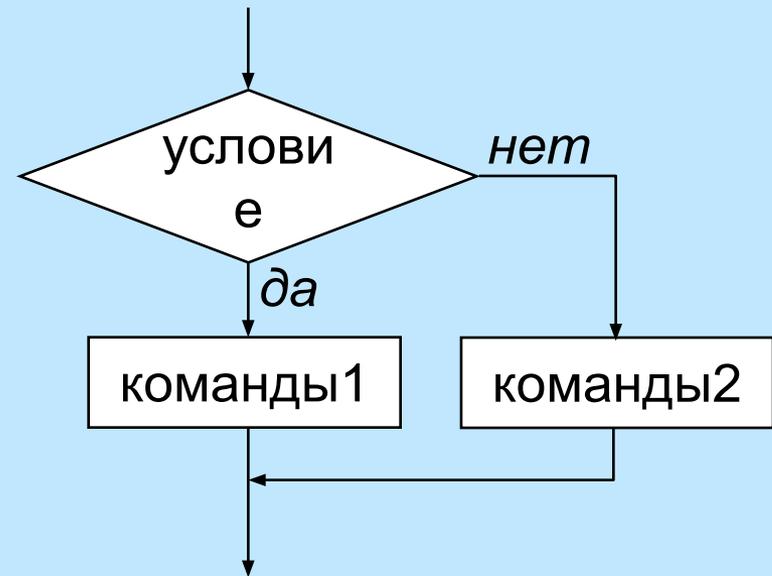
Давайте вспомним

- Линейный алгоритм – это ...
- Разветвляющийся алгоритм – это ...
- Циклический алгоритм – это ...
- Опишите исполнителя Робота по плану:
 - 1) Имя
 - 2) Круг решаемых задач
 - 3) Среда
 - 4) СКИ
 - 5) Система отказов
 - 6) Режимы работы



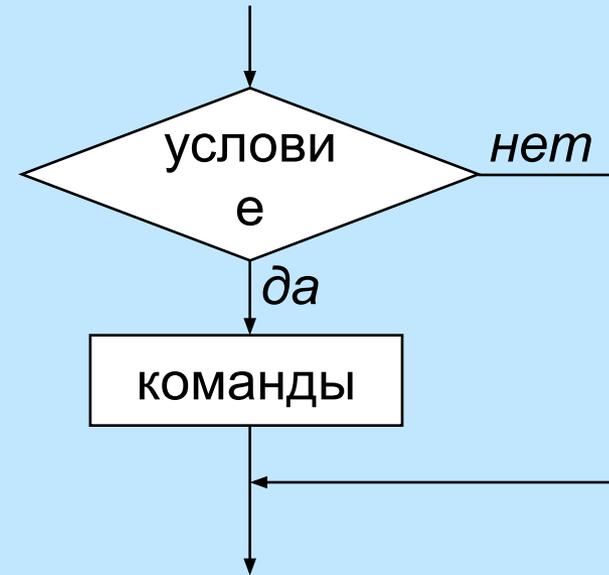
Полная форма ветвления

```
если ( условие )  
  {  
    команды1 ;  
  }  
иначе  
  {  
    команды2 ;  
  }
```



Краткая форма ветвления

```
если ( условие )  
{  
  команды ;  
}
```

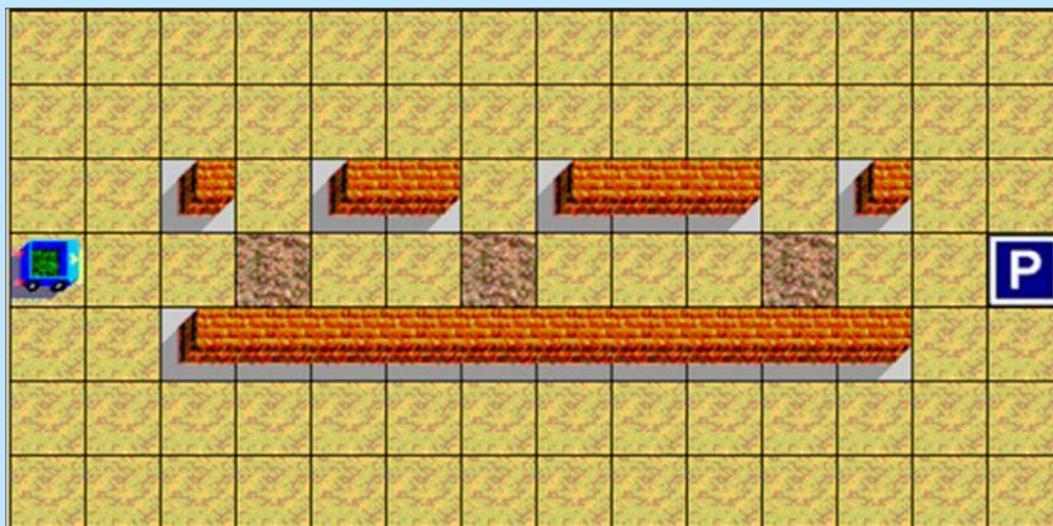


Пример задачи с кратким ветвлением

Робот находится перед коридором неизвестной длины, у которого одна стена сплошная, а другая имеет проходы. Составить программу, после выполнения которой Робот посадит цветы в грядки возле проходов.

Клумбы_выбор

```
{  
  вперед ( 2 );  
  пока ( не справа_свободно )  
  {  
    если ( слева_свободно )  
      { посади; }  
    вперед ( 1 );  
  }  
  вперед ( 1 );  
}
```



Домашнее задание

§3.3 стр. 145-147

РТ № 36, 37, 38 стр. 110-111

Подготовиться к контрольной работе



Дополнительное домашнее задание

(РТ №34 стр. 108, учебник №15 стр.151)

Напишите программу, с помощью которой Робот сможет попасть на Базу во всех трех лабиринтах.

