
**Алгоритмическая
конструкция
«повторение»**

Понятие условия

В жизни человека принятие решений зачастую зависит от различных условий. Если на улице идет дождь, то нужно взять с собой зонтик; если хорошо подготовился к уроку, то получишь высокую отметку, иначе низкую; пока не выполнил домашнее задание, не будешь развлекаться и др.



Понятие условия

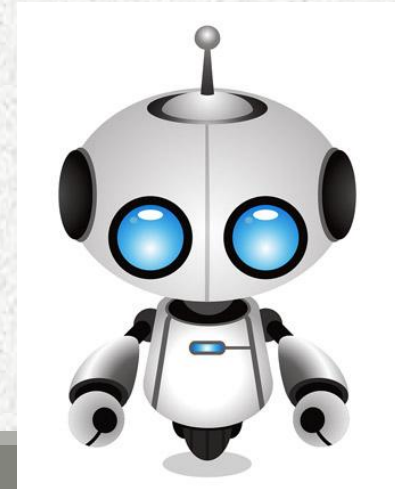
Примеры использования условий

Если горит зеленый свет, можно переходить улицу.



Понятие условия

Человек способен понимать условия, сформулированные в произвольной форме. Но для того чтобы **Робот** или другой исполнитель мог принимать решения, нужно «**научить**» его «**понимать**» **условия**.



Понятие условия

Условием для исполнителя является понятное исполнителю высказывание, которое может быть истинным (соблюдаться) либо быть ложным (не соблюдаться).

Система условий исполнителя Робот

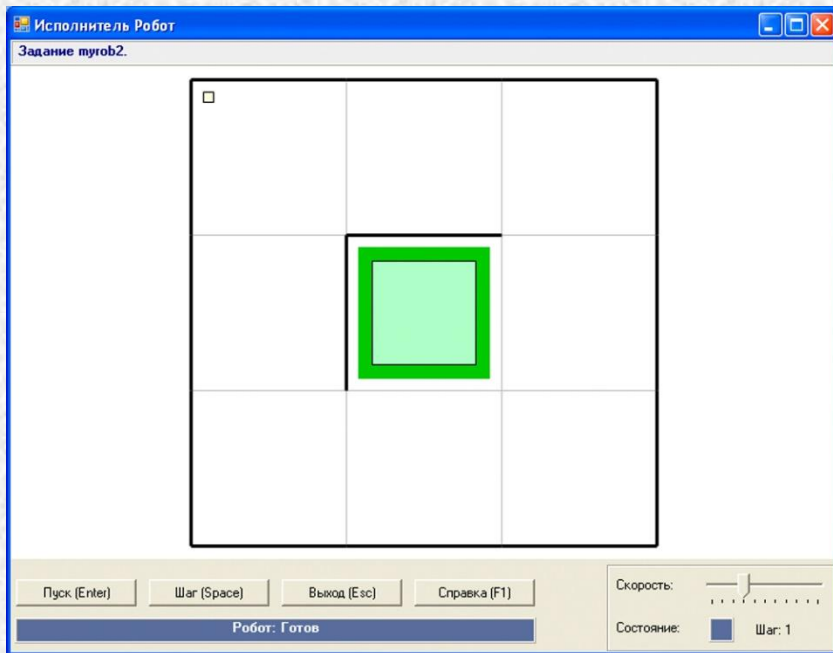
Рассмотрим систему условий для учебного компьютерного исполнителя Робот.

WallFromLeft	Истинно, если слева от Робота стена
WallFromRight	Истинно, если справа от Робота стена
WallFromUp	Истинно, если сверху от Робота стена
WallFromDown	Истинно, если снизу от Робота стена
FreeFromLeft	Истинно, если слева от Робота свободно
FreeFromRight	Истинно, если справа от Робота свободно
FreeFromUp	Истинно, если сверху от Робота свободно
FreeFromDown	Истинно, если снизу от Робота свободно
CellIsPainted	Истинно, если ячейка, в которой находится Робот, закрашена
CellIsFree	Истинно, если ячейка, в которой находится Робот, не закрашена

Образцы истинных и ложных условий для исполнителя Робот

Рассмотрим начальную обстановку поля Робота:

Для Робота будут истинны следующие условия:



WallFromLeft

WallFromUp

FreeFromRight

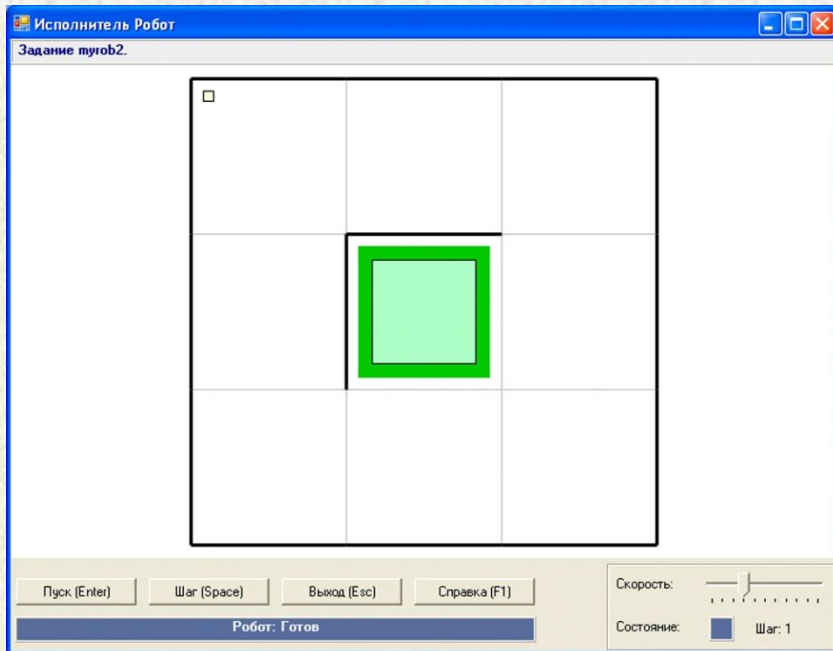
FreeFromDown

CellisPainted

Образцы истинных и ложных условий для исполнителя Робот

Рассмотрим начальную обстановку поля Робота:

Ложными будут условия :



WallFromRight

WallFromDown

FreeFromLeft

FreeFromUp

CellisFree

Цикл с предусловием

Алгоритм собирания грибов



**Пока корзина не будет полная
повторять:**

1. Найти гриб
2. Срезать гриб
3. Положить гриб в корзину.

Цикл с условием

Алгоритм собирания грибов



Цикл с предусловием

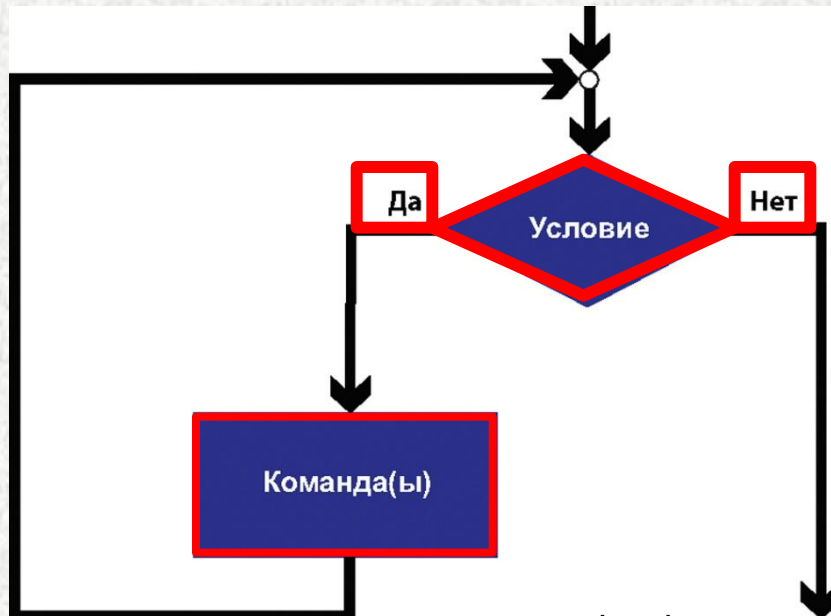
Алгоритмическая конструкция **цикл с предусловием (цикл «пока»)** — способ организации цикла, при котором количество выполнений команд тела цикла зависит от истинности или ложности условия цикла.

Цикл с предусловием

- **Цикл с предусловием** используется в том случае, когда количество повторений тела цикла заранее не известно, но известно условие продолжения работы.
- Условие цикла определяет, как долго будет выполняться цикл.
- Пока условие истинно, выполняются команды, составляющие тело цикла.
- **Цикл** прекращает выполняться тогда, когда условие становится ложным.
- **Цикл с предусловием** имеет такое название, поскольку перед выполнением команд тела цикла проверяется условие.

Цикл с предусловием

Алгоритмическая конструкция цикла с предусловием может изображаться на блок-схеме следующим образом:



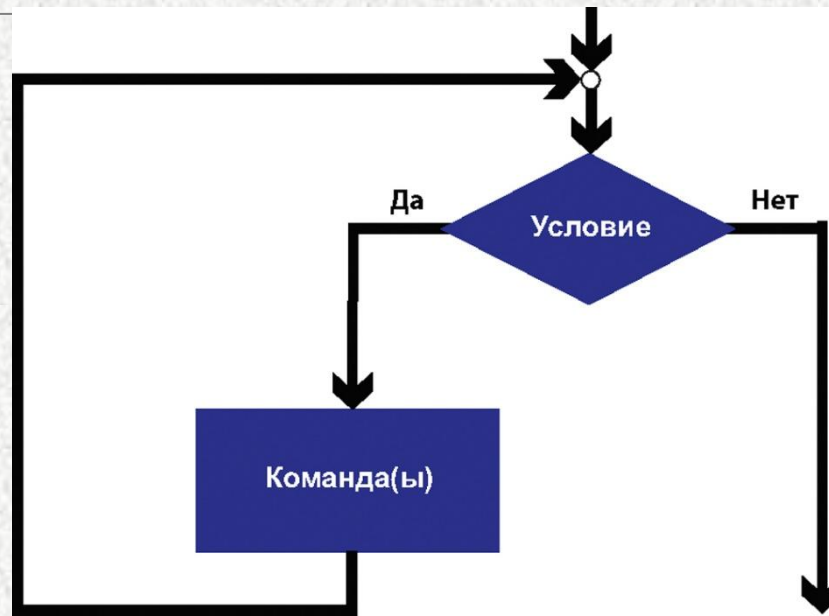
В данной конструкции в прямоугольнике(ах) записываются повторяющиеся команды алгоритма (тело цикла), которые совершаются, пока верно условие (Да).

При этом после каждого выполнения команд тела цикла происходит проверка, истинно ли условие.

Как только условие станет ложным (Нет), цикл завершается.

Цикл с предусловием

Алгоритмическая конструкция цикла с предусловием может изображаться на блок-схеме следующим образом:



Если условие сразу ложно, то цикл не выполнится ни разу.

Если условие в цикле будет всегда истинно (всегда Да), то такой цикл не сможет завершиться. Возникшую ситуацию называют **зацикливанием**.

Цикл с предусловием

Для записи цикла с предусловием используется команда **while**.

Формат записи команды следующий:

```
while <условие> do  
begin  
тело цикла;  
end;
```

Цикл с предусловием

Строка **while** **<условие>** **do** является заголовком цикла. Эту строку можно прочитать следующим образом: **«Пока верно условие, делай»**.

Команды **begin** и **end**; в данном случае играют роль операторных скобок.

```
while <условие> do  
begin  
тело цикла;  
end;
```


Цикл с предусловием

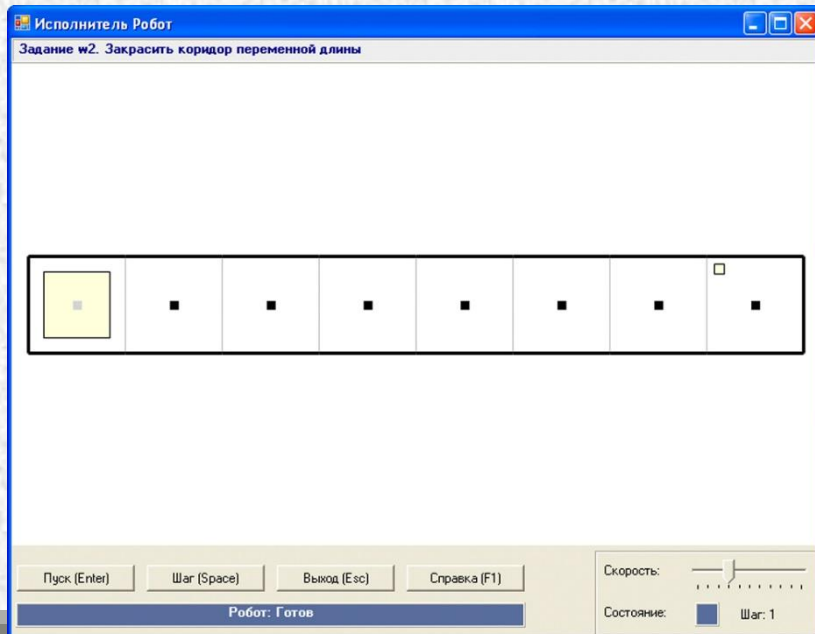
Пример 1

Робот должен закрасить коридор переменной длины.

В данной задаче нам не известна длина коридора, но известно, что **Робот** может двигаться, пока справа пусто, и закрашивать клетки:

Пока справа пусто, **повторять** закрасить; вправо.

После прохода всего коридора **Робот** должен закрасить последнюю клетку.



Цикл с предусловием

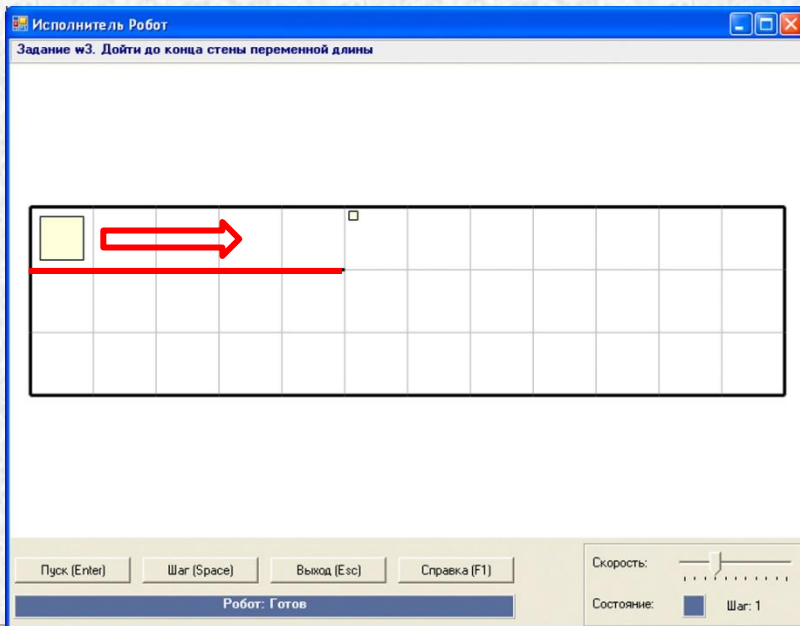
Пример 1

```
uses Robot;  
begin  
Task('w2');  
while FreeFromRight do  
begin  
    paint;  
    right;  
end;  
paint;  
end.
```

Цикл с предусловием

Пример 2

Робот находится в верхнем левом углу поля. Снизу от Робота вдоль поля расположена горизонтальная стена. Составить алгоритм, выполнив который Робот сможет пройти по коридору и остановиться, когда закончится снизу стена. Длина стены заранее не известно.



Мы знаем, что стена снизу не проходит через все поле. **Робот** может двигаться вправо до тех пор, пока внизу есть стена:

Пока снизу стена, повторять вправо

Цикл с предусловием

Пример 2

```
uses Robot;
```

```
begin
```

```
task('w3');
```

```
while WallFromDown do
```

```
right;
```

```
end.
```

Задания 2-8.

Домашнее задание



§12