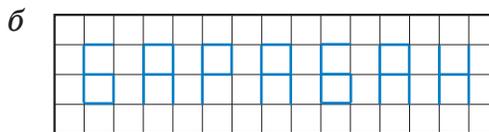
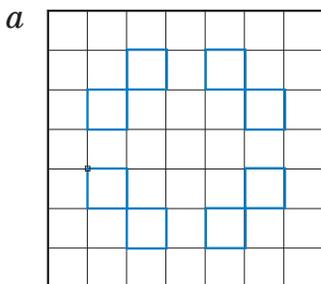
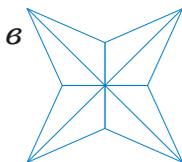
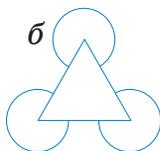
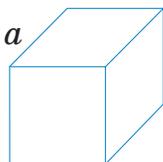


2 Напишите для исполнителя Чертежник программы получения следующих изображений:



3 Придумайте свои рисунки и составьте программы для их рисования с помощью исполнителя Чертежник.

4* Проанализируйте рисунки. Какие из них мог выполнить исполнитель Чертежник? Почему? Какие команды вы можете предложить добавить исполнителю для выполнения остальных рисунков?



§ 9. Исполнитель Робот

9.1. Роботы в жизни человека



Роботы развозят заказы в ресторане в г. Харбин (Китай)¹.

Человек с глубокой древности мечтал об искусственном создании, которое могло бы выполнять его приказы. Сегодня эта мечта стала реальностью — в жизни людей появились роботы. Они способны выполнять практически любую работу, доступ-

¹Материалы о роботах взяты с сайтов <http://www.robogeek.ru> и <http://fishki.net/1211999-roboty-v-nashej-zhizni.html> (дата доступа: 07.02.2017).

ную человеку, а также делать многие вещи, которые людям выполнить сложно или вообще невозможно. Роботы используются на производстве и в быту, могут работать в сфере услуг и развлечений. Есть роботы, похожие на человека, а есть совсем непохожие.

Робот — автоматическое устройство, которое действует по заранее составленной программе.

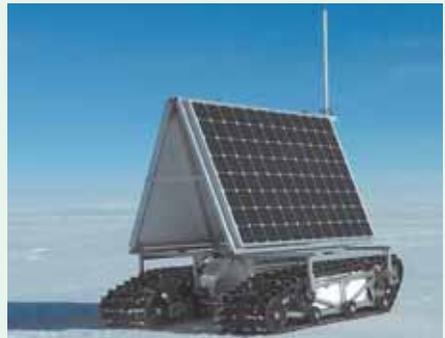
Робот получает информацию о внешнем мире от датчиков — аналогов органов чувств живых организмов — и предназначен для осуществления различных операций.

Мир роботов очень разнообразен. В быту современного человека используются автоматические стиральные и посудомоечные машины, роботы-пылесосы и др. С помощью роботов можно выращивать растения или управлять домом.

Робот может быть материальным или виртуальным. Виртуальный робот — специальная программа, выполняющая определенные действия.



Робот LS3, созданный для транспортировки грузов по пересеченной местности.



Автономный робот GROVER, который изучает слои льда на ледниковом щите Гренландии.

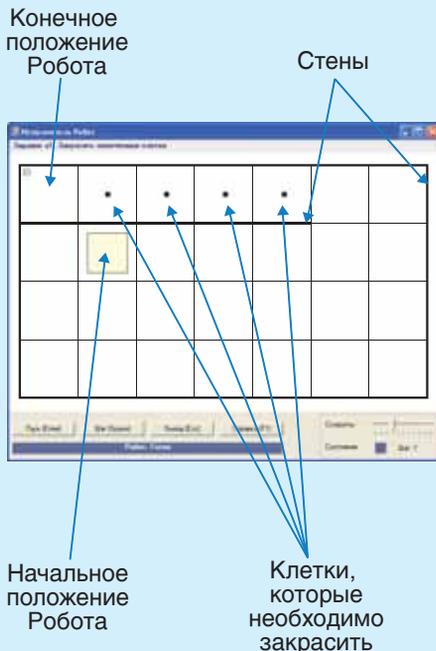


Роботизированная система, предназначенная для выращивания овощей. Управление данной системой осуществляется через Wi-Fi. Есть возможность удаленного контроля через Интернет.



Робот-пылесос с помощью системы камер и сенсоров может ориентироваться в комнате и строить маршрут уборки.

Пример 9.1. Поле исполнителя Робот с начальной обстановкой имеет следующий вид:



Роботы являются исполнителями. Для исполнителей обычно определяют среду обитания и систему команд.

Общим для всех роботов является то, что человек может ими управлять. Робот получает команды от оператора и выполняет их по одной или действует автономно по предварительно составленной программе.

9.2. Среда обитания и система команд исполнителя Робот

В среде программирования PascalABC, кроме исполнителя Чертежник, можно выбрать исполнителя Робот.

Средой обитания исполнителя Робот является прямоугольное клетчатое поле. Размеры поля, как и для исполнителя Чертежник, задаются командой `Field(n, m)`. Первоначально Робот находится в центральной клетке поля.

Между некоторыми клетками, а также по периметру поля находятся стены. Робот может передвигаться по полю и закрашивать указанные клетки. Большой желтый квадрат внутри клетки означает начальное положение Робота, маленький — конечное (пример 9.1).

Поле Робота, на котором определено положение стен, начальное и конечное положение исполнителя, называют **обстановкой**.

Для подключения исполнителя Робот в программе прописывается команда **uses** Robot. Готовые задания с обстановками для Робота хранятся в задачнике, встроенном в систему программирования, и вызываются командой **task**. Эта же команда использовалась для Чертежника.

Система команд исполнителя:

Команда	Действие
Right	Перемещает Робота вправо
Left	Перемещает Робота влево
Up	Перемещает Робота вверх
Down	Перемещает Робота вниз
Paint	Закрашивает текущую ячейку

Робот может становиться на обычную и на закрашенную клетку, но не может переместиться с клетки на клетку, если между ними стена. Робот не может переместиться за границы поля. Эти действия вызывают ошибку (пример 9.2). Робот может закрасить уже закрашенную клетку. Такое действие ошибку не вызывает.

Пример 9.2. Вызов задачи a1 из встроенного задачника:

```

•Program1.pas
uses Robot;
begin
  Task('a1');
end.

```

Запись команд исполнителя:

```

•Program1.pas*
uses Robot;
begin
  Task('a1');
  right;
  right;
end.

```

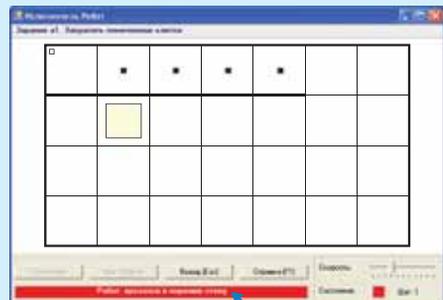
Запишем в программе команду **up**.

```

uses Robot;
begin
  Task('a1');
  up;
end.

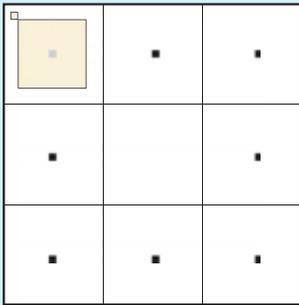
```

Сверху находится стена, поэтому перемещение Робота вверх невозможно:



Сообщение об ошибке

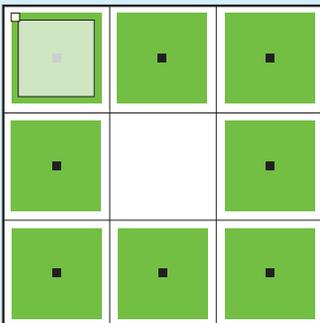
Пример 9.3. Начальная обстановка:



Программа для исполнителя Робот:

```
uses Robot;
begin
  Task('a2');
  paint; right;
  paint; right;
  paint; down;
  paint; down;
  paint; left;
  paint; left;
  paint; up;
  paint; up;
end.
```

Результат работы программы имеет следующий вид:



9.3. Использование алгоритмической конструкции следование для исполнителя Робот

Рассмотрим примеры решения задач для исполнителя Робот.

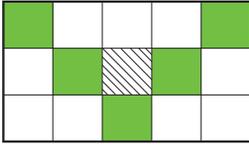
Пример 9.3. Робот находится на поле размером 3×3 клетки. Нужно закрасить все клетки, кроме средней (задача a2).

Для решения задачи Робот должен выполнить следующий алгоритм:

```
закрасить;
вправо;
закрасить;
вправо;
закрасить;
вниз;
закрасить;
вниз;
закрасить;
влево;
закрасить;
влево;
закрасить;
вверх;
закрасить;
вверх.
```

В данном алгоритме Робот обходит клетки, двигаясь по часовой стрелке. Тот же результат можно получить, если Робот будет обходить поле против часовой стрелки, изначально двигаясь вниз.

Пример 9.4. Составим программу для закрашивания клеток поля Робота по образцу:



Такой обстановки нет в задачнике, поэтому вначале нужно создать поле Робота размером 5×3 . Начальное положение Робота на таком поле отмечено заштрихованной клеткой.

Для решения задачи Робот должен выполнить алгоритм:

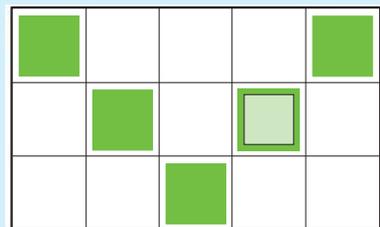
```
создать поле;
вниз;
закрасить;
влево;
вверх;
закрасить;
влево;
вверх;
закрасить;
вправо;
вправо;
вправо;
вправо;
закрасить;
влево;
вниз;
закрасить.
```

** Какими еще способами можно решить данную задачу?*

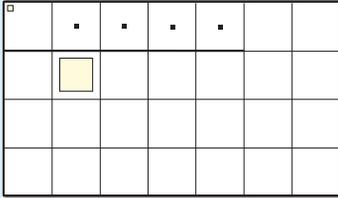
Пример 9.4. В данном случае программа для учебного компьютерного исполнителя Робот может быть составлена таким образом:

```
uses Robot;
begin
  Field(5, 3);
  down;
  paint;
  left;
  up;
  paint;
  left;
  up;
  paint;
  right;
  right;
  right;
  right;
  paint;
  left;
  down;
  paint;
end.
```

Результат работы записанной выше программы будет иметь следующий вид:



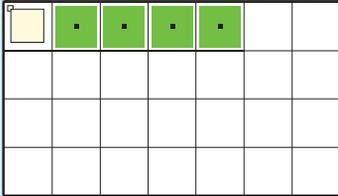
Пример 9.5. Начальная обстановка:



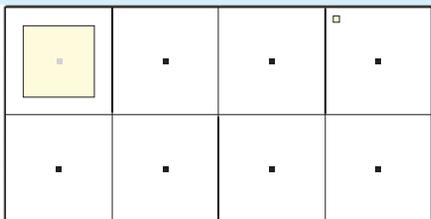
Программа для Робота:

```
uses Robot;
begin
  Task('a1');
  right; right;
  right; right;
  up;
  left; paint;
  left; paint;
  left; paint;
  left; paint;
  left;
end.
```

Результат работы программы:



Пример 9.6. Начальная обстановка:



Пример 9.5. Решим задачу a1 из встроеного задачника.

Для решения данной задачи Робот должен обойти линию (стену), закрасить указанные клетки и переместиться в клетку, определяющую конечное положение исполнителя.

Алгоритм решения задачи:

сдвинуться вправо на 4 клетки;

вверх;

сдвинуться влево на 4 клетки, закрасивая их по пути; влево.

В примерах 9.3 — 9.5 команды исполнителя Робот выполнялись последовательно, одна за другой, в том порядке, в котором они были записаны. Поэтому все приведенные алгоритмы реализованы с использованием алгоритмической конструкции *следование*.

9.4. Вспомогательные алгоритмы

Пример 9.6. Решим задачу a3 из встроеного задачника.

Робот должен закрасить все клетки поля. Но двигаться по прямой ему мешают стены, которые исполнитель должен обходить.

Алгоритм решения задачи:

закрасить; вниз;

закрасить; вправо;

```

закрасить;  вверх;
закрасить;
вправо;
закрасить;  вниз;
закрасить;  вправо;
закрасить;  вверх;
закрасить.

```

Если проанализировать данный алгоритм, то можно заметить, что дважды повторяется последовательность команд, которая закрашивает квадрат из четырех клеток:

```

закрасить;  вниз;
закрасить;  вправо;
закрасить;  вверх;
закрасить.

```

Оформим эти команды как вспомогательный алгоритм, который назовем **квадрат**. Тогда алгоритм решения исходной задачи может быть записан так:

```

квадрат;
вправо;
квадрат.

```

При решении данной задачи использование вспомогательного алгоритма позволило не записывать дважды одну и ту же последовательность команд.

Вспомогательные алгоритмы можно использовать и в том случае, когда исходная задача разбивается на несколько независимых друг от друга задач. Тогда

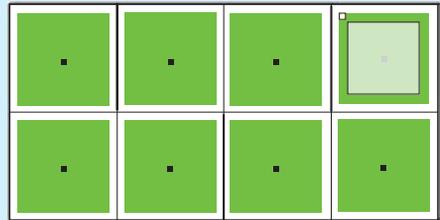
Программа 1 для исполнителя Робот:

```

uses Robot;
begin
  Task('a3');
  paint; down;
  paint; right;
  paint; up;
  paint;
  right;
  paint; down;
  paint; right;
  paint; up;
  paint;
end.

```

Результат работы программы:



Программа 2 (с использованием вспомогательного алгоритма) для исполнителя Робот:

```

uses Robot;
procedure kvadrat;
begin
  paint; down;
  paint; right;
  paint; up;
  paint;
end;
begin
  Task('a3');
  kvadrat;
  right;
  kvadrat;
end.

```

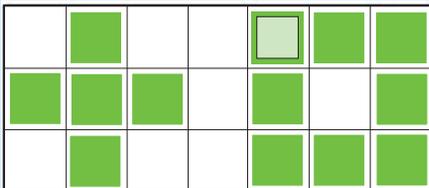
Пример 9.7. Программа для учебного компьютерного исполнителя Робот:

```

uses Robot;
procedure krest;
begin
  left; paint;
  down; left; paint;
  up; left; paint;
  right; paint;
  up; paint;
end;
procedure kvadrat;
begin
  paint; right;
  paint; right;
  paint; down;
  paint; down;
  paint; left;
  paint; left;
  paint; up;
  paint; up;
end;
begin
  field(7,3);
  krest;
  right; right; right;
  kvadrat;
end.

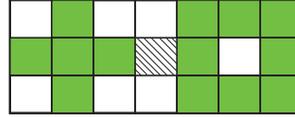
```

Результат работы записанной выше программы будет иметь следующий вид:



каждую из них можно оформить как вспомогательный алгоритм.

Пример 9.7. Напишем программу для закрашивания клеток поля Робота по образцу:



Такой обстановки нет в задачке, поэтому создадим поле 7×3 . Начальное положение Робота отмечено заштрихованной клеткой.

В данной задаче Робот должен нарисовать две отдельные фигуры: крест и квадрат. Составим два вспомогательных алгоритма.

Вспомогательный алгоритм **крест:**

```

влево; закрасить;
вниз; влево; закрасить;
вверх; влево; закрасить;
вправо; закрасить;
вверх; закрасить.

```

В качестве вспомогательного алгоритма для рисования квадрата можно использовать алгоритм решения задачи а2 (пример 9.3). Для перехода от одной фигуры к другой Робот должен сдвинуться на 3 клетки вправо:

```

крест;
вправо; вправо; вправо;
квадрат.

```

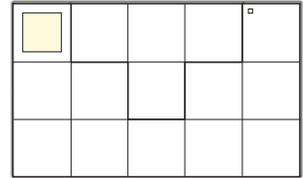


1. Что такое робот?
2. Какие команды входят в систему команд учебного компьютерного исполнителя Робот?
3. Опишите среду обитания учебного исполнителя Робот.

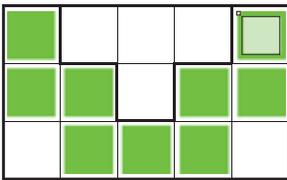


Упражнения

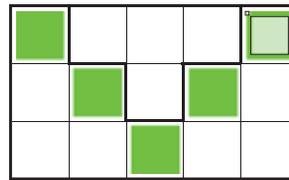
1 Начальная обстановка на поле Робота изображена на рисунке справа. Трое учащихся составили и выполнили алгоритм, по которому Робот закрасил все клетки пути от начальной к конечной. На каком из рисунков — *а*, *б* или *в* — изображено решение данной задачи? Почему?



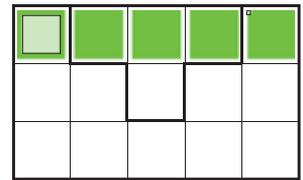
а



б



в



2 Какой из приведенных алгоритмов решает задачу, сформулированную в предыдущем задании? Объясните, почему другие программы не могут быть алгоритмами решения данной задачи.

а) paint; down;
right;
paint; down;
right;
paint; right;
up;
paint; right;
up;
paint;

б) paint; down;
paint; right;
paint; down;
paint; right;
paint; right;
paint; up;
paint; right;
paint; up;
paint;

в) ToPoint (0, 3) ;
PenDown;
OnVector (1, 0) ;
OnVector (0, -1) ;
OnVector (1, 0) ;
OnVector (0, -1) ;
OnVector (1, 0) ;
OnVector (0, 1) ;
OnVector (1, 0) ;
OnVector (0, 1) ;
OnVector (1, 0) ;

3 Для какого исполнителя приведен алгоритм в задании 2, *в*? Сформулируйте для этого исполнителя задачу, решением которой будет приведенный алгоритм.

4 Для исполнителя Робот была составлена следующая программа:

```
paint;
right; up;
paint;
right; down;
```

Изобразите в тетради «узор», который нарисует Робот. При каких минимальных размерах поля Робот сможет выполнить данную программу?

5 Все команды в программе из задания 4 учащийся скопировал три раза. Как изменится «узор» после выполнения программы? Как можно по-другому записать этот алгоритм? Какого размера поле нужно создать? Подсказка. Воспользуйтесь вспомогательным алгоритмом.

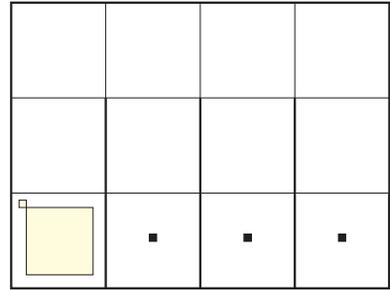
6 Программа решения задачи была записана на доске. Два учащихся, переписывая этот алгоритм для исполнителя Робот, пропустили из-за невнимательности по одной команде. Какую команду пропустил каждый из учащихся? Что будет результатом работы каждой программы?

Программа, записанная первым учащимся	Программа, записанная вторым учащимся
<pre>uses Robot; begin Field(15,15); paint; right; paint; right; paint; down; paint; down; paint; down; left; down; left; paint; down; paint; down; paint; right; paint; right; paint; up; paint; up; paint; up; left; up; left; paint; up; paint; up; end.</pre>	<pre>uses Robot; begin Field(15,15); paint; right; paint; right; paint; down; paint; down; paint; down; left; paint; down; left; paint; down; paint; down; paint; right; paint; right; paint; up; paint; up; paint; up; left; up; paint; up; paint; up; end.</pre>

7 Составьте программу для решения задачи а4 из встроенного задачника (см. рис. справа). Предложите два алгоритма:

1. С использованием алгоритмической конструкции *следование*.
2. С использованием вспомогательного алгоритма.

Сравните полученные решения.



8* Робот-огородник может разбить грядку на посадочные зоны-клетки. На рисунке справа изображена схема посадки овощей (красная клетка — томаты, зеленая — огурцы). Предложите систему команд для робота-огородника и разработайте алгоритм посадки овощей (робот сажает одно растение в одну клетку).



§ 10. Алгоритмическая конструкция *повторение*

10.1. Алгоритмы с циклами

В окружающем мире можно наблюдать ситуации, при которых различные действия и процессы повторяются. Некоторые повторяются несколько раз и завершаются. Другие могут повторяться очень долго (например, круговорот воды в природе, движение планет в космическом пространстве, смена времен года и т. д.). Человеку тоже регулярно приходится выполнять повторяющиеся действия: умываться, одеваться, посещать парикмахерскую, завтракать, ходить на работу и др.

Понятие цикла используется в различных сферах человеческой деятельности.

Под циклом понимают совокупность явлений, процессов, составляющих кругооборот в течение определенного промежутка времени. С этой точки зрения можно говорить о годовом цикле вращения Земли вокруг Солнца или о производственном цикле.

Циклом является законченный ряд каких-либо произведений, чего-либо излагаемого, исполняемого: цикл лекций, цикл стихотворений.