

На бумаге, на классной доске и на экране ЭВМ мы будем изображать поле Робота так, как показано на рисунке 4.

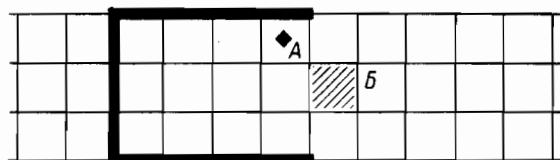


Рис. 4

Робот умеет выполнять всего 17 команд. Мы пока изучим 5: "вверх", "вниз", "вправо", "влево", "закрасить".

По командам "вверх", "вниз", "вправо", "влево" Робот перемещается в соседнюю клетку. В некоторых случаях эти команды могут быть невыполнимы: например, на рисунке 4 выше Робота стена, поэтому команду "вверх" выполнить нельзя.

По команде "закрасить" Робот закрашивает клетку, в которой стоит. Если клетка уже была закрашена, то она будет закрашена еще раз, т. е. останется закрашенной.

#### 4.3. ПРОСТЕЙШИЙ ПРИМЕР АЛГОРИТМА

Пусть требуется перевести Робота из клетки A в клетку B (рис. 4). При управлении "вручную" мы можем два раза скоординировать Робота "вправо" и один раз "вниз". Однако, если мы хотим, чтобы Роботом управляла ЭВМ, мы должны записать эти команды в виде алгоритма на алгоритмическом языке:

```

алг ход конем                                     (A1)
  дано | Робот в клетке A (рис. 4)
  надо | Робот в клетке B (рис. 4)
нач
  вправо
  вправо
  вниз
кон
  
```

#### 4.4. ОБЩИЙ ВИД АЛГОРИТМА

В простейшем случае алгоритм на алгоритмическом языке записывается так:

```

алг имя алгоритма
  дано условия применимости алгоритма
  надо цель выполнения алгоритма
нач
  тело алгоритма (последовательность команд)
кон
  
```

Слова алг (алгоритм), дано, надо, нач (начало), кон (конец) называются *служебными словами* и служат для оформления алгоритма. Часть алгоритма до служебного слова нач называют *заголовком* алгоритма, а часть между словами нач и кон — *телом* алгоритма. Имя (название) алгоритма может быть любым. Обычно оно подбирается так, чтобы можно было понять, для чего предназначен алгоритм. Служебные слова алг, нач и кон пишутся строго одно под другим, нач и кон соединяются вертикальной чертой, правее которой пишется тело алгоритма — последовательность команд.

#### 4.5. КОММЕНТАРИИ В АЛГОРИТМИЧЕСКОМ ЯЗЫКЕ

В алгоритме "ход конем" (A1) после знака | в строках дано и надо записан комментарий. Такие комментарии разрешается помещать в конце любой строки, отделяя их знаком |. Если комментарий занимает несколько строк, то знак | перед комментарием надо писать в каждой строке. Комментарии могут записываться в любой удобной для человека форме. При выполнении алгоритма ЭВМ полностью пропускает комментарии — алгоритм выполняется так же, как если бы комментариев вообще не было. Таким образом, комментарии предназначены исключительно для человека — они облегчают понимание алгоритма.

#### 4.6. ВЫЗОВ КОМАНДЫ ИСПОЛНИТЕЛЯ

При выполнении алгоритма ЭВМ последовательно командует Робота выполнять записанные в алгоритме команды. Каждый такой приказ на выполнение команды называется *вызовом* этой команды.

#### 4.7. ОШИБКИ В АЛГОРИТМАХ

Если при составлении алгоритма мы случайно вместо "вниз" напишем "внис" или вместо "вправо" напишем "направо", то ЭВМ нашу запись не поймет и, даже не приступая к выполнению алгоритма, сообщит об ошибке. Такие ошибки в записи алгоритма называются *синтаксическими*.

Ошибки другого вида — *отказы* — проявляются при выполнении алгоритма. Например, при попытке выполнить алгоритм "ход конем" (A1) в обстановке, изображенной на рисунке 5, ЭВМ попытается последовательно вызвать команды "вправо", "вправо",

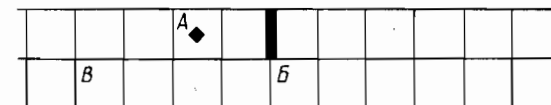


Рис. 5

"вниз". Однако вторую команду "вправо" Робот выполнить не сможет. Такие ситуации называются отказами. Если при выполнении алгоритма возникнет отказ, то ЭВМ сообщит об ошибке и прекратит выполнение.

Кроме синтаксических ошибок и отказов, в алгоритме могут быть **логические** ошибки, не обнаруживаемые ЭВМ ни до выполнения алгоритма, ни при его выполнении. Например, если в алгоритме А1 мы вместо "вправо" случайно напишем "влево", то ЭВМ выполнит алгоритм, Робот из клетки А (рис. 5) переместится в клетку В (рис. 5), но никаких сообщений об ошибках мы не получим (да и откуда ЭВМ знать, куда мы на самом деле хотели переместить Робота!).

В правильно составленных алгоритмах никаких ошибок быть не должно. Но если синтаксические ошибки и отказы обычно легко устранимы, то поиск и устранение логических ошибок может оказаться весьма трудным делом.

#### 4.8. ЗАПИСЬ НЕСКОЛЬКИХ КОМАНД В ОДНОЙ СТРОКЕ

Правила алгоритмического языка разрешают записывать в одной строке несколько команд через точку с запятой. Пусть требуется перевести Робота из клетки А в клетку В (рис. 6, а).

Путь, который должен пройти Робот, можно разбить на пять одинаковых участков (рис. 6, б). Команды прохождения каждого участка можно сгруппировать в одну строку — это сокращает запись алгоритма и делает его более понятным:

**алг** из А в Б (A2)

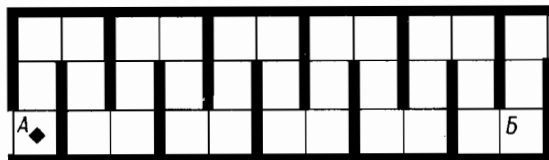
**дано** | Робот в клетке А (рис. 6, а)

**надо** | Робот в клетке Б (рис. 6, а)

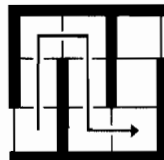
**нач**

вверх; вверх; вправо; вниз; вниз; вправо  
вверх; вверх; вправо; вниз; вниз; вправо  
вверх; вверх; вправо; вниз; вниз; вправо  
вверх; вверх; вправо; вниз; вниз; вправо  
вверх; вверх; вправо; вниз; вниз; вправо

**кон**



а)



б)

Рис. 6

#### УПРАЖНЕНИЯ

1. Дан алгоритм, в котором стерты комментарии и название:

а) **алг** (A3) б) **алг** (A4)

**дано** |  
**надо** |  
**нач**  
вверх; закрасить; вниз  
вправо; закрасить; влево  
вниз; закрасить; вверх  
влево; закрасить; вправо  
**кон**

**дано** |  
**надо** |  
**нач**  
вверх; вправо; закрасить  
вниз; вниз; закрасить  
влево; влево; закрасить  
вверх; вверх; закрасить  
вправо; вниз  
**кон**

Опишите движение Робота в процессе выполнения алгоритма. Нарисуйте конечное положение Робота и закрашенные в результате выполнения клетки. Придумайте подходящее название алгоритма и впишите комментарии после слов **дано** и **надо**.

2. Измените алгоритм задачи 1, б) так, чтобы при его исполнении Робот:

а) прошел тем же маршрутом, но ничего не закрашивал;

б) закрасил все клетки, в которых он побывал.

3. Известно, что на поле Робота нет стен и закрашенных клеток. Определите, сколько клеток будет закрашено после исполнения следующих команд:

а) закрасить	б) закрасить
вправо	вправо
вверх	закрасить
закрасить	закрасить
вправо	вправо
закрасить	вправо
вверх	закрасить
закрасить	закрасить
закрасить	закрасить
вправо	вправо

4. Известно, что на поле Робота имеется одна стена, равная по длине стороне клетки, а попытка выполнить последовательность команд из задачи 3, а) приводит к отказу. Измените последовательность команд так, чтобы конечное положение Робота осталось прежним, закрашивались те же клетки, а отказа не возникало.

5. Составьте алгоритм, при выполнении которого Робот переместится из клетки А в клетку Б (рис. 7).

6. Приведите все алгоритмы из трех команд, переводящие Робота из клетки А в клетку Б в задаче 5, а.

7. Существует ли алгоритм для задачи 5, а, при выполнении которого Робот делает: а) два шага; б) четыре шага; в) семь шагов; г) 1990 шагов?

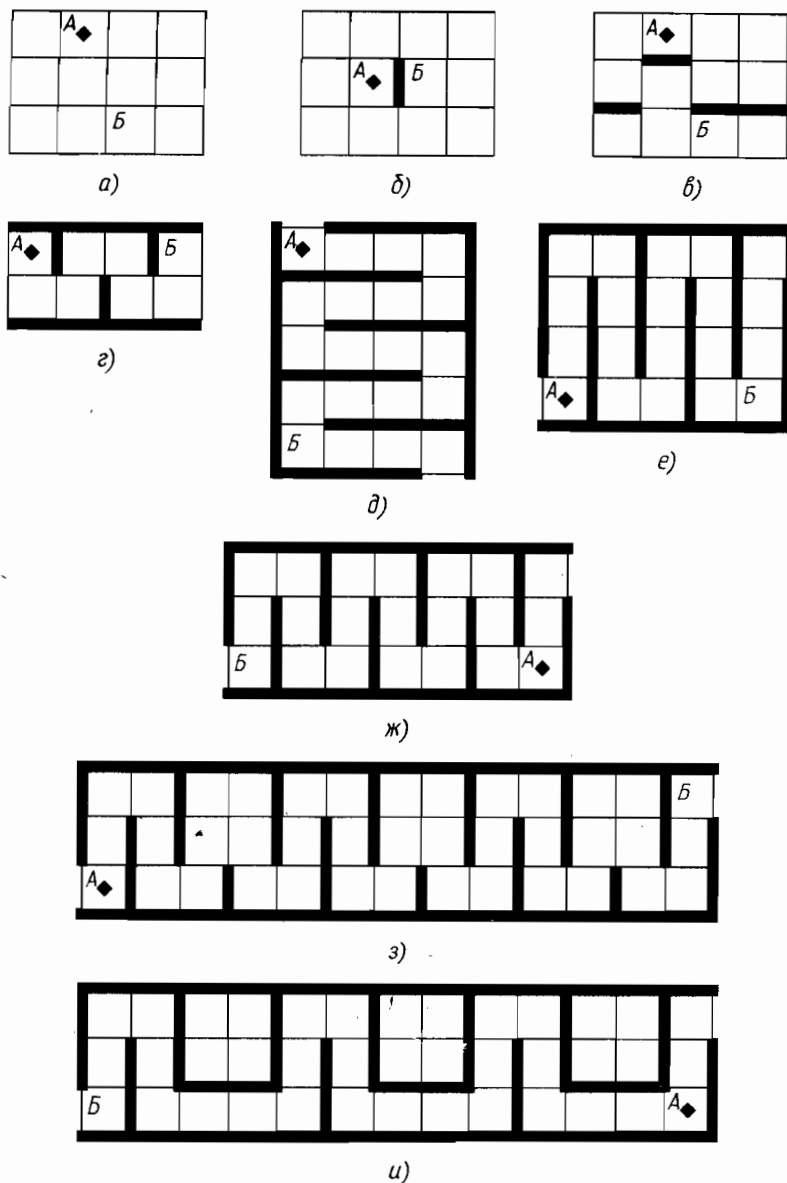


Рис. 7

8. Составьте алгоритм, который перемещает Робота из А в Б и закрашивает клетки, отмеченные точками (рис. 8).

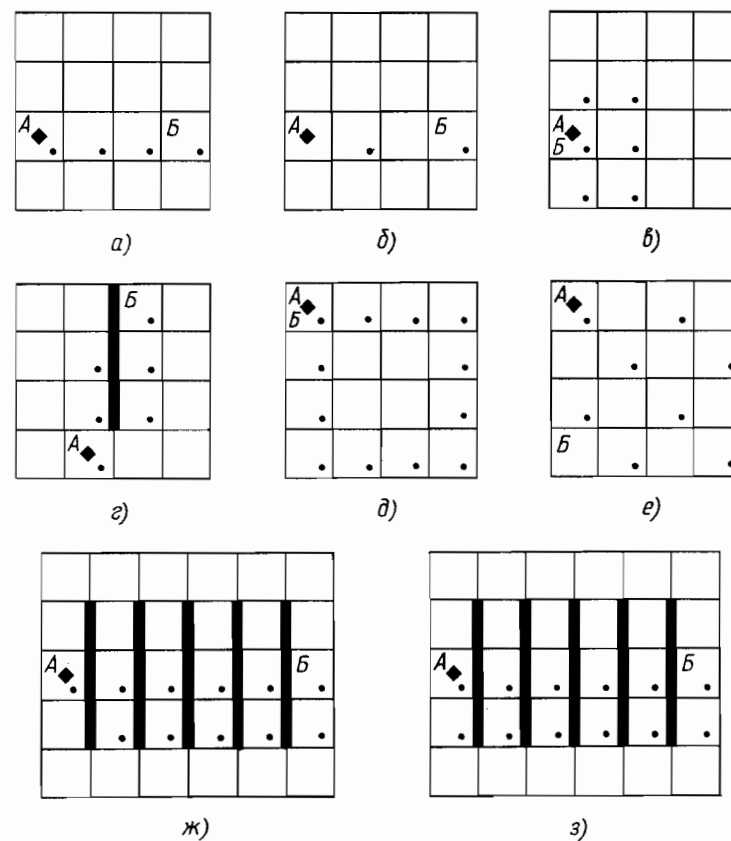


Рис. 8

7. Петя составил алгоритм, а Коля стер в нем одну команду:

алг прогулка (А5)  
дано | стен на поле нет  
надо | Робот погулял и вернулся в исходное положение  
нач  
 | вверх  
 | вправо  
 | ???  
 | вниз  
 | влево  
 | влево  
кон

Определите, что за команду стер Коля, если известно, что при выполнении составленного Петей алгоритма Робот возвращался в исходное положение.

8. Петя составил алгоритм, при выполнении которого Робот вернулся в исходное положение. Коля стер одну из команд. При выполнении Колиного алгоритма Робот также вернулся в исходное положение. Какую команду стер Коля?

9. Петя составил алгоритм, при выполнении которого на поле без стен Робот вернулся в исходное положение. Коля переставил две команды местами. Докажите, что при выполнении Колиного алгоритма Робот также вернется в исходное положение.

10. Петя составил алгоритм для Робота, который на поле без стен и закрашенных клеток закрашивает 5 клеток. Коля переставил в алгоритме две соседние команды. Может ли новый алгоритм закрашивать а) 3; б) 4; в) 5; г) 6; д) 7 клеток?

11. Петя составил алгоритм, при выполнении которого Робот закрашивает 5 клеток. Коля переставил в алгоритме какие-то две команды (необязательно соседние). Может ли новый алгоритм закрашивать а) 0; б) 1; в) 5; г) 7; д) 100 клеток?

12. Петя составил алгоритм, переводящий Робота из клетки А в клетку Б с закрашиванием каких-то клеток. Что должен сделать Коля с этим алгоритмом, чтобы получить алгоритм, переводящий Робота из Б в А и закрашивающий те же клетки?

## § 5. ИСПОЛНИТЕЛЬ «ЧЕРТЕЖНИК» И РАБОТА С НИМ

### 5.1. ОСОБЕННОСТИ ЗАПИСИ ЧИСЕЛ В ИНФОРМАТИКЕ

В информатике для отделения целой части числа от дробной используется точка, а не запятая. Это позволяет записывать несколько рядом стоящих чисел через запятую без риска вызвать путаницу. При задании точек плоскости координаты  $x$  и  $y$  в информатике разделяются запятой (рис. 9).

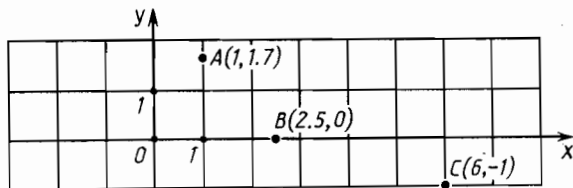


Рис. 9

### 5.2. ИСПОЛНИТЕЛЬ «ЧЕРТЕЖНИК»

**Чертежник** предназначен для построения рисунков, чертежей, графиков и т. д. на бесконечном листе бумаги. Чертежник имеет перо, которое можно поднимать, опускать и перемещать.

При перемещении опущенного пера за ним остается след — отрезок от старого положения пера до нового. Всего Чертежник умеет выполнять 4 команды:

поднять перо  
опустить перо  
сместиться в точку (**арг вещь**  $x, y$ )  
сместиться на вектор (**арг вещь**  $a, b$ )

Слова **арг** (*аргументы*) и **вещ** (*вещественные*) указывают, что команды "сместиться в точку" и "сместиться на вектор" имеют аргументы, которые могут быть произвольными вещественными числами. Аргументы указываются при вызове команды, например: "сместиться в точку (2.5,0.5)".

### 5.3. РАБОТА КОМАНДЫ «СМЕСТИТЬСЯ НА ВЕКТОР»

Если перо Чертежника находится в точке  $(x, y)$ , то по команде "сместиться на вектор  $(a, b)$ " Чертежник сместит перо в точку с координатами  $(x+a, y+b)$  (рис. 10, а). Если перо опущено, то при этом будет нарисован отрезок от старого положения пера до нового — от точки  $(x, y)$  до точки  $(x+a, y+b)$ . Этот отрезок является изображением вектора с координатами  $(a, b)$  и началом в точке  $(x, y)$ , чем и объясняется название команды.

Если, например, перо Чертежника опущено и находится в точке  $(1, 2)$ , то при выполнении команды "сместиться на вектор  $(3, 3)$ " Чертежник сместит перо в точку  $(1+3, 2+3)$ , т. е. нарисует отрезок от точки  $(1, 2)$  до точки  $(4, 5)$  (рис. 10, б).

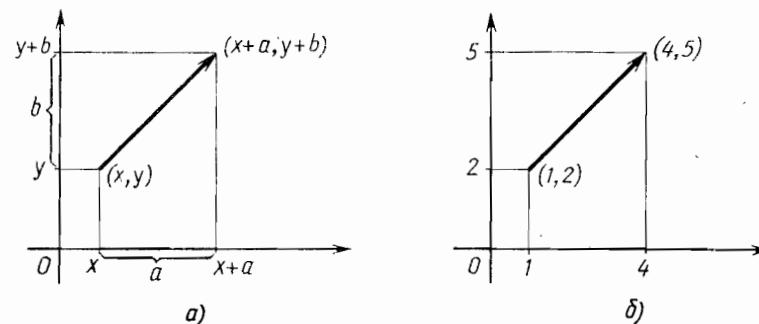


Рис. 10

### 5.4. ПРИМЕР АЛГОРИТМА УПРАВЛЕНИЯ ЧЕРТЕЖНИКОМ

Попробуем с помощью Чертежника нарисовать прямоугольник с двумя диагоналями (рис. 11, а). Как это сделать? Можно начать с левого нижнего угла и, двигаясь против часовой стрелки,