

# **Основные команды языка Лого**

# Содержание

1. Команды движения
2. Команды поворотов
3. Команды пера черепахи
4. Процедуры
5. Использование процедур в процедурах
6. Процедуры с параметрами

# Команды движения

Самая простая команда это команда, заставляющая черепашку двигаться **вперед**, а двигается черепашка туда, куда направлена ее голова. При загрузке проекта голова первой черепашки смотрит вверх и поэтому двигаться она по этой команде будет вверх.

Почти у всех команд есть параметры. Это числа, которые уточняют последующие действия черепашки. Так и у команды «вперед» есть параметр. Он означает, сколько шажков должна сделать черепашка. Так как ножки у черепашки очень коротенькие, то 10 шажков черепашки это приблизительно одна клеточка школьной тетради. Вот как выглядит, например, команда, которая заставит черепашку прошагать 100 шажков:

**вперед 100**

# Команды движения

Чтобы не писать длинные слова, команды можно сокращать. Так, команду **вперед** можно сократить до двух букв – **вп**. Таким образом, чтобы отправить черепашку в короткое путешествие, достаточно дать ей команду **вп 100**.

Аналогичная команда заставляет черепашку пятиться назад: **назад 100** или в сокращенной форме - **нд 100**.

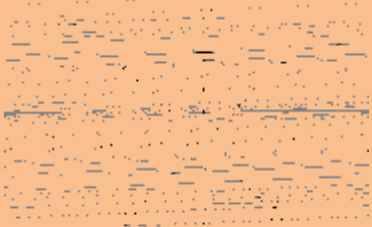
# Команды поворотов

Понятно, что двигаться только по одной прямой для черепашки не очень интересно. Поэтому она может поворачиваться. Повернуть черепашку можно командами направо и налево:

<i>Команда</i>	<i>Параметр</i>	<i>Действие</i>	<i>Сокращенная</i>	<i>Примеры</i>
<b>направо</b>	Количество градусов	Черепашка поворачивается направо относительно текущего состояния на указанное количество градусов	<b>пр</b>	<b>направо 45 пр 90</b>
<b>налево</b>	Количество градусов	Черепашка поворачивается налево относительно текущего состояния на указанное количество градусов	<b>лв</b>	<b>налево 90 лв 60</b>

# Команды поворотов

Иногда трудно вспомнить текущее положение головы черепашки, но надо установить ее в какое-то строго определенное положение, например строго вверх, или налево. Тогда удобнее использовать команду **новый курс**.

<i>Команда</i>	<i>Параметр</i>	<i>Действие</i>	<i>Сокращенная</i>	<i>Примеры</i>
<b>нов_курс</b>	 Количество градусов, отсчитываемых от нулевого положения черепахи головой вверх по часовой стрелке	Черепашка поворачивается в соответствии с указанным направлением	<b>нк</b>	<b>нов_курс 90</b> <b>нк 225</b>

# Команды пера черепахи

Черепашка при движении может оставлять следы, т. е. рисовать пером. При открытии нового проекта перо черепашки поднято, т.е. черепашка при движении не оставляет следов, но если дать команду «**перо опусти**», то черепашка при движении будет оставлять след.

<i>Команда</i>	<i>Параметр</i>	<i>Действие</i>	<i>Сокращенная</i>	<i>Примеры</i>
<b>по</b>	нет	Черепашка «опускает перо». После этой команды, куда бы она не двигалась, будет оставлять след.	—	<b>по</b>
<b>пп</b>	нет	Черепашка «поднимает перо», т.е. после этой команды она будет двигаться бесследно.	—	<b>пп</b>

# Команды пера черепахи

Цвет пера черепашки при открытии нового проекта устанавливается черным (номер 9), а толщина равна 1 (самое тонкое перо). Эти характеристики пера можно поменять описанными ниже командами:

<i>Команда</i>	<i>Параметр</i>	<i>Действие</i>	<i>Сокращенная</i>	<i>Примеры</i>
<b>нов_размер_пера</b>	Размер пера в условных единицах. Самый маленький размер равен 1, самый большой 100.	Черепашка меняет свое перо в соответствии с указанным размером	<b>нрп</b>	<b>нрп 5</b>
<b>нов_цвет</b>	Либо название, либо номер цвета. Номер можно узнать, если в графическом редакторе навести на нужный цвет мышку. Всплывет подсказка с номером цвета.	Цвет пера черепашки меняется на указанный	<b>нц</b>	<b>нов_цвет 15</b> <b>нц “красный”</b>



# Процедуры

**Процедура - это часть программы, выполняющая самостоятельную задачу и реализующая тот или иной алгоритм, который можно применить в другой процедуре или программе.**

Таким образом, процедуры до бесконечности расширяют набор примитивов (команд) для черепашек, ведь любая написанная вами программа может считаться еще одной командой.

# Процедуры

Первая строка любой процедуры должна содержать слово “это” и имя процедуры. Предпочтительней процедуру называть одним словом. Нежелательно применять в имени разделители и знаки препинания. Количество знаков в имени тоже лучше ограничить в связи с тем, что при вызове процедуры можно сделать ошибки при написании длинных имен. Итак, предлагается начать знакомство с темой написанием процедуры рисования правильной фигуры, например, квадрата.

# Процедуры

Текст процедуры записывается на листе программ для Лого миров (*Ctrl + F*). Последней строкой в процедуре должна быть строка «**конец**». Это важно, потому что на листе программ может быть написано много процедур. Признаком окончания процедуры является слово «конец». Все, что написано между строками «конец» и следующей процедурой, начинающейся со слова «это» воспринимается в ЛОГО как комментарий

# Команда повторения

Для того, чтобы написать простейшую процедуру (программу) изучим еще одну команду. Это команда, заставляет черепашку повторять то или иное действие какое-то количество раз. Таким образом, это примитив (команда) реализует циклическую структуру алгоритма

<i>Команда</i>	<i>Параметр</i>	<i>Действие</i>	<i>Сокращенная</i>	<i>Примеры</i>
<b>повтори <i>k</i> [действия]</b>	<i>k</i> – количество повторений; <i>д</i> <i>ействия</i> – команды или определенные пользователем процедуры.	Черепашка повторяет те действия, которые указаны в скобках	—	<b>повтори 100 [вп 1 жди 1]</b> черепашка будет медленно двигаться 100 шагов (жди это пауза в действиях)

# Процедуры

Теперь можно приняться за обучение черепашки новым командам. Первая из них будет заставлять черепашку рисовать квадрат. Алгоритм рисования очень простой. Черепашка должна идти какое-то количество шагов вперед, а потом поворачиваться на  $90^{\circ}$ .

Примерный текст процедуры квадрата следующий:

```
это квадрат  
по  
повтори 4[ вп 60 пр 90]  
конец
```

Для того, чтобы черепашка выполнила новую команду, достаточно просто написать имя процедуры в поле команд:

**квадрат**

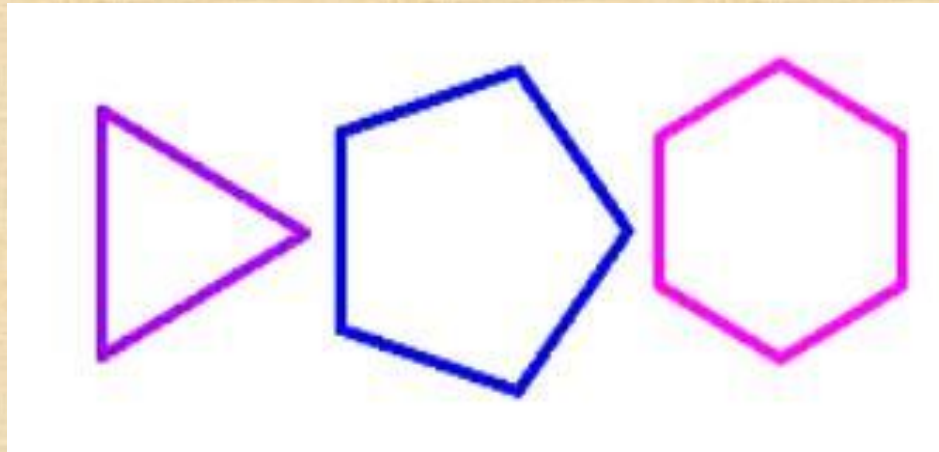
# Процедуры

Теперь у черепашки есть новая команда, которую она может выполнять.

## Упражнение 1

Создайте процедуры рисования равносторонних треугольника, пятиугольника, шестиугольника. Чем отличаются эти процедуры?

В своих процедурах предусмотрите увеличение толщины пера и изменения цвета рисования (см. команды **нрп** и **нц**).



# Использование процедур в процедурах

Созданные пользователем процедуры используются также, как и обычные команды для черепашек. Попробуем, например, использовать созданную ранее процедуру для рисования узора:

Этот узор составлен из шести квадратов, повернутых относительно друг друга на  $60^{\circ}$ . Текст программы узора может быть таким:

```
это узор  
нрп 4 нц 25 по  
повтори 6[квадрат пр 60]  
конец
```

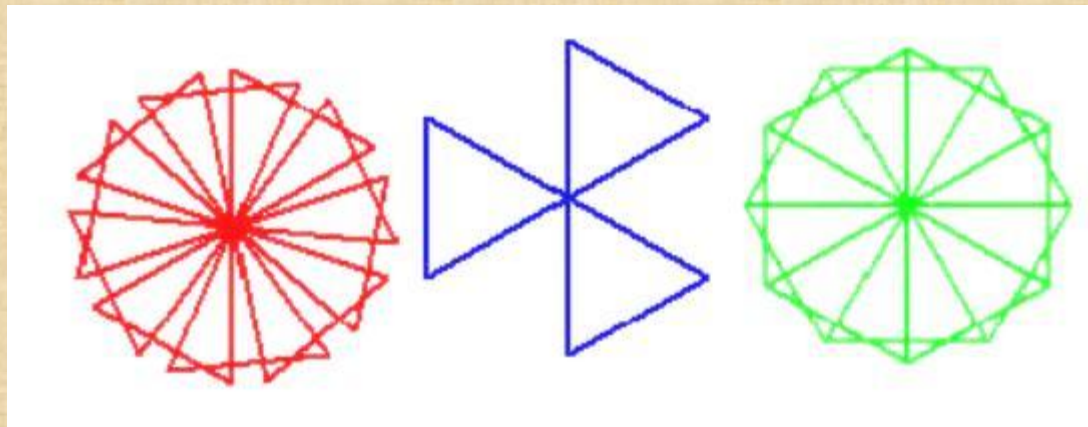


При выполнении процедуры узор черепашка нарисует похожую на приведенный рисунок картинку.

# Использование процедур в процедурах

## Упражнение 2

Создайте проект, где в листе программ определите процедуру рисования треугольников. Попробуйте создать три узора (в виде процедур на листе программ), использующие процедуру рисования треугольника, изображенные на рисунке.





# Процедуры с параметрами

Мы научились писать программы на листе программ и заставлять черепашку выполнять придуманные нами процедуры. Сейчас попробуем сделать свои процедуры гибче и универсальнее.

Умение определять нужные параметры и правильно их задавать, очень важно при работе с процедурами в программировании.

# *Процедуры с параметрами*

**Параметр это величина, которая может принимать различные значения и описывает какие-либо характеристики объектов. В модуле параметр имеет имя, записываемое чаще всего буквами. Значение параметра можно изменить, при этом меняется результат работы процедуры.**

# Процедуры с параметрами

В качестве примера можно рассмотреть процедуры для квадрата и треугольника. Введем для них один параметр, означающий длину стороны правильной фигуры. Обозначим его :д. В Лого знак «:» перед именем параметра обязателен и показывает, что мы имеем дело со значением параметра, а не с его именем. При этом не стоит забывать, что между «:» и именем параметра пробела быть не должно.

Имена всех используемых параметров нужно обязательно указывать в первой строке процедуры (в заголовке процедуры). С учетом всего вышесказанного тексты процедур для квадрата и треугольника будут выглядеть следующим образом:

# Процедуры с параметрами

\*\*\*\*\*

**это кв :д**

**по**

**повтори 4[ вп :д пр 90]**

**конец**

\*\*\*\*\*

**это тр :д**

**по**

**повтори 3[вп :д пр 120]**

**конец**

\*\*\*\*\*

Для того, чтобы черепашка нарисовала нужные фигуры, при вызове процедур необходимо указать конкретные значение параметра. Например, для получения квадрата со стороной 100 шагов черепахи надо набрать в поле команд строку:

**кв 100**

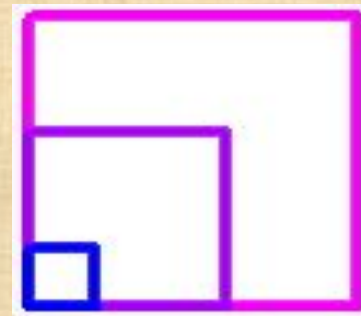
При выполнении этой команды параметру :д присваивается значение, равное 100 и выполняется процедура с этим значением параметра.

# Процедуры с параметрами



## Упражнение 1

Попробуйте нарисовать подобную картинку с помощью созданной вами процедуры с параметром для рисования квадратов.

Используйте команды *нц* и *нри* для задания цвета и толщины линии. Оформите отдельную процедуру для рисования этой картинку на листе программ



# Процедуры с параметрами

Создайте кнопку на своем проекте с помощью инструмента . Для этого выберите этот инструмент и растяните на проекте прямоугольник, определяющий размер и положение будущей кнопки. Откройте диалоговое окно кнопки с помощью инструмента  (нажмите на глаз, а затем на кнопку) и введите в окно команд название созданной вами процедуры для рисования узора. После этого подтвердите ввод, нажав кнопку ОК.

# Процедуры с параметрами

Теперь при нажатии на эту кнопку активная черепашка будет рисовать узор.

Аналогичную кнопку сделайте для похожего узора из треугольников.

Что пришлось вам делать каждый раз после рисования квадрата или треугольника в узорах? Попробуйте ввести второй параметр в процедуры рисования квадратов и треугольников - цвет пера черепашки. Если у процедуры несколько параметров, то они и при задании и при вызове разделяются пробелом. Отредактируйте процедуры узоров для новых процедур.



такой рисунок, применим созданные нами процедуры с параметрами для рисования прямоугольников и треугольников.

# Процедуры с параметрами

Примерный вид вашего проекта:

В качестве примера рассмотрим рисунок лица, составленного из простых геометрических фигур. Для того чтобы получить такой рисунок, применим созданные нами процедуры с параметрами для рисования прямоугольников и треугольников.

