

**Муниципальное бюджетное общеобразовательное учреждение  
«Быковская основная общеобразовательная школа  
Яковлевского городского округа»**

<p>РАССМОТРЕНО на методическом совете МБОУ "Быковская ООШ" Протокол №1 от «28» августа 2024 г.</p>	<p>СОГЛАСОВАНО зам. директора МБОУ «Быковская ООШ»  Турчина С.В. от «29» августа 2024 г.</p>	<p>УТВЕРЖДЕНО директор МБОУ "Быковская ООШ"  Осадчая Н.Г. Приказ № 278 от «29» августа 2024 г.</p>
--	---	---

**Рабочая программа курса внеурочной деятельности  
«Основы программирования»  
(5-6 класс)  
Срок реализации рабочей программы– 1 год**

**Автор:  
Турчина Светлана Викторовна**

**с.Быковка 2024г.**

Сегодня человеческая деятельность в технологическом плане меняется очень быстро, на смену существующим технологиям и их конкретным техническим воплощениям быстро приходят новые, которые специалисту приходится осваивать заново. В этих условиях велика роль фундаментального образования, обеспечивающего профессиональную мобильность человека, готовность его к освоению новых технологий, в том числе информационных. Поэтому в содержании курса информатики основной школы целесообразно сделать акцент на изучении фундаментальных основ информатики, выработке навыков алгоритмизации, реализовать в полной мере общеобразовательный потенциал этого курса.

Программа внеурочной деятельности «Основы программирования» позволит расширить кругозор обучающихся. Посредством формирования начальных навыков программирования готовится платформа для изучения более сложных языков и тем.

Для изучения основ программирования в 5-6 классах был выбран язык программирования Logo (Лого) в среде Kturtle. Творческая среда Лого имеет много возможностей для формирования у детей начальной компьютерной грамотности, навыков работы с вычислительной техникой, развития алгоритмического и логического мышления, познавательной грамотности. В настоящее время этот язык включает в себя множество разнообразных команд, позволяющих рисовать, решать вычислительные задачи, создавать мультфильмы и многое другое. Лого привлекает и тем, что программирование из скучной дисциплины превращается в интересную игру, в процессе которой происходит быстрое усвоение основных понятий и навыков программирования.

**Цель:** Обучение учащихся основам программирования, развитие их алгоритмического мышления.

### **Основные задачи курса:**

- освоить среду Лого и стандартные команды исполнителя Черепашки;
- освоить понятие «алгоритм» и изучить виды и свойства алгоритма.
- Развить творческие способности и логическое мышление учащихся.
- Приобщить учащихся к проектной деятельности с использованием различных средств работы в среде Лого.
- Развить навыки самостоятельного поиска решений.

## Планируемые результаты

В результате освоения курса «Основы программирования» в 5-6 классах учащиеся получают представление:

- о понятии «информация» — одном из основных обобщающих понятий современной науки, о понятии «данные», о базовых понятиях, связанных с хранением, обработкой и передачей данных;
- о методах представления и алгоритмах обработки данных, дискретизации, о программной реализации алгоритмов;
- о математических и компьютерных моделях, их использовании,
- о различных видах программного обеспечения и задачах, решаемых с его помощью; о существовании вредоносного программного обеспечения и средствах защиты от него, о необходимости стандартизации в сфере информационно-коммуникационных технологий;

У учащихся будут сформированы:

- основы алгоритмической культуры;
- умение составлять несложные программы;
- навыки и умения, необходимые для работы с основными видами программных систем;
- навыки коммуникации с использованием современных средств ИКТ, включая непосредственное выступление перед аудиторией;
- представления о необходимости учёта юридических аспектов использования ИКТ, о нормах информационной этики.

Обучающиеся познакомятся с одним из языков программирования и основными алгоритмическими структурами — линейной, условной и циклической; получат опыт написания и отладки программ в выбранной среде программирования.

*Учащийся научится:*

- понимать термины «исполнитель», «состояние исполнителя», «система команд»; понимать различие между непосредственным и программным управлением исполнителем;
- строить модели различных устройств и объектов в виде исполнителей, описывать возможные состояния и системы команд этих исполнителей;
- понимать термин «алгоритм»; знать основные свойства алгоритмов (фиксированная система команд, пошаговое выполнение, детерминированность, возможность возникновения отказа при выполнении команды);
- составлять неветвящиеся (линейные) алгоритмы управления исполнителями и записывать их на выбранном алгоритмическом языке (языке программирования);
- использовать логические значения, операции и выражения с ними;
- понимать (формально выполнять) алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы), вспомогательных алгоритмов, простых и табличных величин;
- создавать алгоритмы для решения несложных задач, используя конструкции ветвления (условные операторы) и повторения (циклы), вспомогательные алгоритмы и простые величины;
- создавать и выполнять программы для решения несложных алгоритмических задач в выбранной среде программирования.

*Выпускник получит возможность:*

- познакомиться с использованием в программах строковых величин и с операциями со строковыми величинами;
- создавать программы для решения задач, возникающих в процессе учебы и вне её;
- познакомиться с задачами обработки данных и алгоритмами их решения;
- познакомиться с понятием «управление», с примерами того, как компьютер управляет различными системами (летательные и космические аппараты, станки, оросительные системы, движущиеся модели и др.)

**Учебно-тематический план.**

№ п.п	Тема	Кол-во часов
	<b>Алгоритмы и исполнители.</b>	<b>3</b>
1.	Правила техники безопасности. Алгоритмы и исполнители.	1
2.	Свойства алгоритмов. Способы записи алгоритмов.	1
3.	Виды алгоритмов.	1
	<b>Введение в Kturtle.</b>	<b>3</b>
1.	Запуск программы. Интерфейс программы.	1
2.	Подготовка к работе. Настройка программы.	1
3.	Последовательность работы.	1
	<b>Программирование на языке Лого.</b>	<b>26</b>
1.	Первая программа. Команды перемещения.	1
2.	Команды очистки. Управление спрайтом.	1
3.	Практическая работа № 1 «Черепашка рисует дом».	1
4.	Сохранение проектов. Практическая работа № 2 «Рисуем кораблик».	1
5.	Управление пером черепашки.	1
6.	Практическая работа № 3 «Рисуем цветной домик».	1

7.	Работа с холстом.	1
8.	Практическая работа № 4 «Изменение параметров холста».	1
9.	Переменные в Лого. Контейнеры.	1
10.	Практическая работа № 5 «Работа с переменными».	1
11.	Текстовые контейнеры.	1
12.	Практическая работа № 6 «Склеиваем слова».	1
13.	Получение случайных чисел.	1
14.	Практическая работа № 7 «Работа со случайными числами».	1
15.	Вывод данных на экран. Организация диалога.	1
16.	Практическая работа № 8 «Организация диалога с черепашкой».	1
17.	Условный оператор.	1
18.	Практическая работа № 9 «Нахождение большего из двух чисел».	1
19.	Сложные условия.	1
20.	Практическая работа № 10 «Использование сложных условий».	1
21.	Повторение команд. Цикл со счетчиком.	1
22.	Практическая работа № 11 «Рисуем правильные многоугольники».	1
23.	Цикл с условием.	1

24.	Практическая работа № 12 «Рисуем узоры».	1
25.	Подпрограммы.	1
26.	Практическая работа № 13 «Учим черепашку новым командам».	1
1.	<b>Создание собственных проектов.</b>	<b>2</b>
1.	<b>Защита проектов.</b>	<b>1</b>

## Содержание программы.

### 1. Алгоритмы и исполнители.

Правила техники безопасности и организация рабочего места. Алгоритмы и исполнители. Свойства алгоритмов (массовость, дискретность, результативность, детерминированность, понятность и выполнимость). Способы записи алгоритмов (словесный, блок-схема, программа). Виды алгоритмов (линейный, разветвляющийся, циклический).

### 1. Введение в Kturtle.

Запуск программы Kturtle различными способами (через меню и терминал). Интерфейс программы (строка меню, редактор кода, панель инструментов, холст, строка состояния, исполнитель команд — черепашка). Знакомство с меню программы.. Подготовка к работе. Настройка программы (настройка языка и размера рабочего поля). Последовательность работы (запись программы, запуск сценария, сообщение об ошибках, отладка программы, скорость выполнения программы, редактирование программы).

### 1. Программирование на языке Лого.

Команды перемещения (вперед X, назад X, налево X, направо X). Команды очистки (очисти, сброс). Управление спрайтом (спрячь, покажи). Сохранение проектов с использованием меню. Управление пером черепашки (поднять перо,

опустить перо, ширина пера, цвет пера). Работа с холстом (размер холста, цвет холста, обертка). Переменные в Лого. Контейнеры (переменные, пустые контейнеры, значение переменной, операция присвоения, числовое значение переменной, арифметические выражения). Текстовые контейнеры (строки, сложение строк или «склеивание», результат сложения строк). Получение случайных чисел (случайное число, диапазон чисел, случайное число из диапазона чисел). Вывод данных на экран (команда напиши, размер шрифта, пиксель). Организация диалога (обмен данными с черепашкой через диалог, окно вопроса, команда «сообщение»). Условный оператор (последовательность действий, условие, выбор последовательности действий в зависимости от условия, команда если-то-иначе, логическое выражение). Сложные условия (сложные условия, состоящие из простых логических выражений и логических операций *и*, *или* и *не*, таблицы истинности). Повторение команд (повторение последовательности команд несколько раз). Цикл со счетчиком (цикл *для*, тело цикла, рисование правильных многоугольников). Цикл с условием (цикл *пока*, условие выполнения цикла). Подпрограммы (подпрограммы, неограниченные команды черепашки, создание новых команд черепашки, входные параметры, группы операторов).

### **1. Создание собственных проектов.**

Разработка собственного проекта по предложенным темам.

### **1. Защита проектов.**

Защита собственного проекта.