

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Комитет по образованию Правительства Санкт-Петербурга  
Администрация Санкт-Петербурга  
Отдел образования Пушкинского района Санкт-Петербурга  
ГБОУ школа № 315

РАССМОТРЕНО

Педагогический совет

Протокол №1 от 28.08.2024 г.

УТВЕРЖДАЮ

Директор \_\_\_\_\_ /А. А. Миренкова/

Приказ №83 от 28.08.2024 г.

РАБОЧАЯ ПРОГРАММА

Внеурочной деятельности «Программирование»

для обучающихся 9 классов

Санкт-Петербург

2024-2025

## **ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

Рабочая программа курса внеурочной деятельности «Программирование» (далее Программа) является составной частью основной образовательной программы основного общего образования ГБОУ школы №315.

Программа составлена в соответствии с федеральными, региональными и муниципальными нормативными документами, перечень которых представлен в качестве приложения к основной образовательной программе основного общего образования ГБОУ школы №315.

Программа курса внеурочной деятельности «Программирование» является закономерным продолжением уроков информатики. Программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта основного общего образования.

Рабочая программа «Программирование» представляет собой вариант программы внеурочной деятельности по общеинтеллектуальному направлению для учащихся 9 классов, которая способствует созданию условий для интеллектуального развития ребенка и формирования его коммуникативных и социальных навыков через практическую, проектную и исследовательскую деятельность, посредством использования информационных технологий.

Курс построен таким образом, чтобы помочь учащимся заинтересоваться программированием и найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации; научиться общаться с компьютером, который ничего не сможет сделать, если человек не напишет для него соответствующую программу.

Данная программа внеурочной деятельности в условиях ФГОС ООО предназначена учителям основного общего образования, педагогам дополнительного образования.

### **Направленность программы**

Программа технической направленности, которая направлена на:

- формирование и развитие творческих способностей обучающихся
- развитие и поддержку детей, проявивших интерес и определенные способности к техническому творчеству, техническому моделированию.

### **Новизна программы**

Задача построения в стране новой инновационной экономики и достижения технологического уровня, запланированного Концепцией долгосрочного социально-экономического развития Российской Федерации и долгосрочным прогнозом научно-технологического развития Российской Федерации до 2025 года, не может быть решена без существенных изменений системы дополнительного образования детей, создания новых общеразвивающих программ технической направленности.

Изменение взглядов на программирование как науку, его место в системе

научного знания требует существенных изменений в содержании образовательного процесса. В связи с этим особую актуальность приобретают раскрытие личностных резервов учащихся и создание соответствующей образовательной среды.

Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения в отношении к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции – одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных учащихся. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в формеразличных программ.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей.

Содержание обучения, представленное в программе «Программирование», позволяет вести обучение в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов – блок-схем, алгоритмов, программ – способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у учащихся знаний.

### **Актуальность программы**

С развитием современных информационных технологий сегодня любой учащийся под руководством опытного педагога может с лёгкостью научиться программировать. Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями. Языки программирования можно сравнить с иностранными языками, овладеть ими может каждый. Учиться программировать очень интересно. Результат программирования очень часто виден сразу. Кроме того, создание компьютерных игр и обучающих программ способствует развитию логики и креативного мышления. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

Потребностью общества в специалистах, владеющих профессиональными навыками и умением. Определение и выбор учащихся (ещё на стадии школьного обучения) дальнейшего профессионального развития, обучения и освоения конкретных специальностей, более лёгкой адаптацией «во взрослой» жизни.

Актуальность программы проявляется в том, что на современном этапе развития общества она отвечает запросам детей и родителей: формирует социально значимые знания, умения и навыки оказывает комплексное обучающее,

развивающее, воспитательное и здоровьесберегающее воздействие, способствует формированию эстетических и нравственных качеств личности, приобщает детей к творчеству.

### **Адресат программы**

Курс «Программирование» рассчитан на учеников 9 классов, проявляющих интерес к информационно-коммуникационным технологиям.

### **Объем и срок освоения программы**

Количество учебных часов – 34 ч (1 ч в неделю). Продолжительность реализации программы: 1 учебный год.

**Формы обучения** - очное, с применением электронного обучения, дистанционных образовательных технологий. Занятия ведутся на русском языке

### **Педагогическая целесообразность**

Состоит в том, что активизация познавательного процесса позволяет учащимся более полно выразить свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению информационных компетенций в других учебных курсах, а также способствует возникновению дальнейшей мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

Курс служит средством внутри профильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

Целесообразность программы выражена в подборе интерактивных и практико-ориентированных форм занятий, способствующих формированию основных компетенций (информационных, коммуникативных, компетенций личного развития и др.).

В рамках изучения курса программы «Программирование» – это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков. Особую роль в формировании отводится формированию мыслительных и психических процессов учащихся (внимание, память, логика), освоение приёмов умственных действий, самостоятельного нахождения и составления алгоритмов решения задач, умения строить модели, чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования предметных и метапредметных результатов учащихся особенно важно, т.к. именно они активизируют процесс индивидуально-личностного становления учащихся.

Основное количество часов отводится практическому написанию программ. Каждый обучающийся реализует индивидуальный проект. Продукт, полученный в результате освоения программы, имеет прикладной характер и может быть

использован по необходимости.

Можно без преувеличения сказать, что программа имеет потенциальные предпосылки для развития коммуникативных и речевых действий в силу их действительно универсального, т. е. максимально обобщенного, характера.

Заложенные в Федеральном государственном образовательном стандарте второго поколения основы формирования универсальных учебных действий подчёркивают ценность современного образования – школа должна побуждать молодёжь принимать активную гражданскую позицию, усиливать личностное развитие и безопасную социальную включённость в жизнь общества.

**Цель программы:** способствовать развитию инженерного мышления, воспитание конкурентно способной личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

### **Задачи**

#### **Образовательные:**

- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- обучить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.
- обучить языку программирования Python и созданию программ на его основе;
- научить создавать прикладное программное обеспечение;
- расширять кругозор обучающихся в области программирования;
- научить дизайнерскому оформлению созданного ПО.

#### **Личностные результаты**

- критическое отношение к информации и избирательность её восприятия;
- осмысление мотивов своих действий при выполнении заданий;
- развитие любознательности, сообразительности при выполнении разнообразных заданий проблемного и эвристического характера;
- развитие внимательности, настойчивости, целеустремленности, умения преодолевать трудности – качеств весьма важных в практической деятельности любого человека;
- развитие самостоятельности суждений, независимости и нестандартности мышления;
- воспитание чувства справедливости, ответственности;
- начало профессионального самоопределения, ознакомление с миром профессий, связанных с робототехникой.

#### **Метапредметные результаты**

### ***Регулятивные универсальные учебные действия:***

- принимать и сохранять учебную задачу;
- планировать последовательность шагов алгоритма для достижения цели;
- формировать умения ставить цель – создание творческой работы, планировать достижение этой цели;
- осуществлять итоговый и пошаговый контроль по результату;
- адекватно воспринимать оценку учителя;
- различать способ и результат действия;
- вносить коррективы в действия в случае расхождения результата решения задачи на основе ее оценки и учета характера сделанных ошибок;
- в сотрудничестве с учителем ставить новые учебные задачи;
- проявлять познавательную инициативу в учебном сотрудничестве;
- осваивать способы решения проблем творческого характера в жизненных ситуациях;
- оценивать получающийся творческий продукт и соотносить его с изначальным замыслом, выполнять по необходимости коррекции либо продукта, либо замысла.

### ***Познавательные универсальные учебные действия:***

- осуществлять поиск информации в индивидуальных информационных архивах учащегося, информационной среде образовательного учреждения, в федеральных хранилищах информационных образовательных ресурсов;
- использовать средства информационных и коммуникационных технологий для решения коммуникативных, познавательных и творческих задач;
- ориентироваться на разнообразие способов решения задач;
- осуществлять анализ объектов с выделением существенных и несущественных признаков;
- проводить сравнение, классификацию по заданным критериям;
- строить логические рассуждения в форме связи простых суждений об объекте;
- устанавливать аналогии, причинно-следственные связи;
- моделировать, преобразовывать объект из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);
- синтезировать, составлять целое из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов;
- выбирать основания и критерии для сравнения и классификации объектов;

### ***Коммуникативные универсальные учебные действия:***

- аргументировать свою точку зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;
- выслушивать собеседника и вести диалог;
- признавать возможность существования различных точек зрения и права каждого иметь свою;
- планировать учебное сотрудничество с учителем и сверстниками — определять цели, функции участников, способов взаимодействия;
- осуществлять постановку вопросов — инициативное сотрудничество в поиске и сборе информации;

- разрешать конфликты – выявление, идентификация проблемы, поиск и оценка альтернативных способов разрешения конфликта, принятие решения и его реализация;
- управлять поведением партнера — контроль, коррекция, оценка его действий;
- уметь с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации;
- владеть монологической и диалогической формами речи.

## Учебный план

| №<br>п/п  | Наименование<br>разделов и<br>тем   | Количество часов |          |          | Формы аттестации/<br>контроля                  |
|-----------|---|------------------|----------|----------|--|
|           |   | Всего            | Теория   | Практика |  |
| <b>1.</b> | <b>Раздел 1. Знакомство с языком Python</b>   | <b>4</b>         | <b>2</b> | <b>2</b> | Тестирование,<br>решение практических<br>задач |
| 1.1       | Общие сведения о языке<br>Практическая работа: Установка<br>Программы Python                                      | 2                | 1        | 1        |  |
| 1.2.      | Режимы работы<br>Практическая работа: Режимы работы<br>с Python   | 2                | 1        | 1        |  |
| <b>2.</b> | <b>Раздел 2. Переменные и выражения</b>   | <b>8</b>         | <b>4</b> | <b>4</b> | Тестирование,<br>решение практических<br>задач |
| 2.1       | Переменные<br>Практическая работа: Работа со<br>справочной системой   | 2                | 1        | 1        |  |
| 2.2       | Практическая работа: Переменные   | 1                |          | 1        |  |
| 2.3       | Выражения<br>Практическая работа: Выражения   | 2                | 1        | 1        |  |
| 2.4       | Ввод и вывод  | 1                | 1        |          |  |
| 2.5       | Задачи на элементарные действия с<br>числами<br>Практическая работа: Задачи на<br>элементарные действия с числами | 2                | 1        | 1        |  |
| <b>3</b>  | <b>Раздел 3. Условные предложения</b>   | <b>8</b>         | <b>4</b> | <b>4</b> | Тестирование,<br>решение практических<br>задач |
| 3.1       | Логические выражения и операторы.<br>Практическая работа: Логические<br>выражения                                 | 2                | 1        | 1        |  |
| 3.2       | Условный оператор<br>Практическая работа: Условный оператор   | 2                | 1        | 1        |  |
| 3.3       | Множественное ветвление<br>Практическая работа: Множественное<br>ветвление  | 2                | 1        | 1        |  |
| 3.4       | Реализация ветвления в языке Python.<br>Практическая работа: «Условные<br>операторы»                              | 2                | 1        | 1        |  |
| <b>4</b>  | <b>Раздел 4. Циклы</b>  | <b>10</b>        | <b>5</b> | <b>5</b> | Тестирование,<br>решение практических<br>задач |
| 4.1.      | Оператор цикла с условием<br>Практическая работа Числа<br>Фибоначчи   | 2                | 1        | 1        |  |
| 4.2.      | Оператор цикла for<br>Практическая работа Решение задачи с<br>циклом for.   | 2                | 1        | 1        |  |
| 4.3.      | Вложенные циклы<br>Практическая работа: Реализация<br>циклических алгоритмов                                      | 2                | 1        | 1        |  |
| 4.4.      | Случайные числа<br>Практическая работа: Случайные числа   | 2                | 1        | 1        |  |



|          |  |           |           |           |                            |
|----------|--|-----------|-----------|-----------|----------------------------|
| 4.5.     | Примеры решения задач с циклом<br>Практическая работа: Решение задач с циклом.       | 2         | 1         | 1         |                            |
| <b>5</b> | <b>Раздел 5. Строки -<br/>последовательности символов</b>                            | <b>4</b>  | <b>1</b>  | <b>3</b>  | Решение практических задач |
| 5.1.     | .Строки<br>Практическая работа: Строки   | 1         |           | 1         |                            |
| 5.2.     | Срезы строк  | 2         | 1         | 1         |                            |
| 5.3      | Примеры решения задач со строками<br>Практическая работа: Решение задач со строками. | 1         |           | 1         |                            |
|          | <b>ВСЕГО</b>   | <b>34</b> | <b>16</b> | <b>18</b> |                            |

### Календарно-тематическое планирование «Программирование»

| Дата занятия | Корректировка дат | Кол-во часов | Содержание занятия   | Раздел, тема  | Время на изучение |              |
|--------------|-------------------|--------------|--|---|-------------------|--------------|
|              |                   |              |  |   | теоретических     | практических |
|              |                   | 2            | <b>Практические занятия.</b> Общие сведения о языке Python. Установка Python на компьютер.   | Общие сведения о языке Python<br>Установка программы Python | 1                 | 1            |
|              |                   | 2            | <b>Теоретические занятия.</b> Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python.<br><b>Практические занятия</b><br>Программы Python. Режимы работы с Python | Режимы работы<br><br>Режимы работы с Python                 | 1                 | 1            |
|              |                   | 2            | Переменные. Оператор присваивания. Имена переменных и ключевые слова.<br><b>Практические занятия</b> Выражения. Операции. Порядок выполнения операций.   | Работа со справочной системой                               | 1                 | 1            |
|              |                   | 1            | <b>Практические занятия</b> Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.   | Практическая работа:<br>Переменные                          |                   | 1            |

|  |  |   |  |  |   |   |
|--|--|---|--|--|---|---|
|  |  | 2 | <b>Теоретические занятия</b> Выражения. Операции.<br><b>Практические занятия</b> Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами | Выражения<br>Практическая работа:<br>Выражения | 1 | 1 |
|  |  | 1 | <b>Теоретические занятия</b> Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных                                | Ввод и вывод                                   |   | 1 |
|  |  | 2 | <b>Теоретические занятия.</b> Решение задач на элементарные действия с числами.<br><b>Практические занятия</b> Выполнение практической работы                                | Задачи на элементарные действия с числами      | 1 | 1 |

|  |  |   |   |   |   |   |
|--|--|---|---|---|---|---|
|  |  | 2 | <b>Теоретические занятия</b> Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not).<br><b>Практические занятия</b> Примеры решения задач с условным оператором. Множественное ветвление. | Логические выражения и операторы. Практическая работа: Логические выражения | 1 | 1 |
|  |  | 2 | <b>Теоретические занятия</b> Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not).<br><b>Практические занятия</b> Работа с программными блоками.  | Условный оператор<br>Практическая работа:<br>Условный оператор              | 1 | 1 |

|  |  |   |   |   |   |   |
|--|--|---|---|---|---|---|
|  |  | 2 | <p><b>Теоретические занятия</b> Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.</p> <p><b>Практические занятия</b> Примеры решения задач с условным оператором. Множественное ветвление</p> | Множественное ветвление   | 1 | 1 |
|  |  | 2 | <p><b>Практические занятия</b> Работа с программными блоками</p>  | <p>Реализация ветвления в языке Python</p> <p>Практическая работа: «Условные операторы»</p> |   | 2 |

|  |  |   |  |   |   |   |
|--|--|---|--|---|---|---|
|  |  | 2 | <p><b>Практические занятия</b> Числа Фибоначчи<br/>Практическая работа</p>   | Практическая работа Числа Фибоначчи                             |   | 2 |
|  |  | 2 | <p><b>Теоретические занятия</b> Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах</p> <p><b>Практические занятия</b> Реализация циклических алгоритмов<br/>Практическая работа</p> | <p>Вложенные циклы</p> <p>Реализация циклических алгоритмов</p> | 1 | 1 |
|  |  | 2 | <p><b>Практические занятия</b> Реализация циклических алгоритмов<br/>Практическая работа</p>   | Реализация циклических алгоритмов                               |   | 2 |
|  |  | 2 | <p><b>Теоретические занятия</b> Случайные числа. Функция randrange. Функция random.</p>  | Случайные числа   | 2 |   |

|  |  |   |  |                                   |   |   |
|--|--|---|--|-----------------------------------|---|---|
|  |  | 2 | <b>Практические занятия</b><br>Решения задач со случайными числами   | Случайные числа                   |   | 2 |
|  |  | 1 | <b>Теоретические занятия</b> Составной тип данных-строка. Доступ по индексу. Длина строки и отрицательные индексы.<br>Преобразование типов. Применение цикла для обхода строки.<br><b>Практические занятия</b><br>Решение практических задач, создание строк | Строки                            |   | 1 |
|  |  | 2 | <b>Теоретические занятия</b><br>Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов<br><b>Практические занятия</b><br>Решение практических задач, создание строк                                      | Срезы строк                       | 1 | 1 |
|  |  | 1 | <b>Теоретические занятия</b> Разбор примеров решения задач со строками   | Примеры решения задач со строками |   | 1 |

## Содержание учебного плана

### Раздел 1. Знакомство с языком Python (4 часа)

Теория: Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Практическая работа 1.1. Установка программы Python  
Практическая работа 1.2. Режимы работы Python

### Раздел 2. Переменные и выражения (8 часов)

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Работа со справочной системой

Практическая работа 2.2. Переменные

Практическая работа 2.3. Выражения

Практическая работа 2.4. Задачи на элементарные действия с числами

### Раздел 3. Условные предложения (8 часов)

Теория: Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3.1. Логические выражения

Практическая работа 3.2. Условный оператор

Практическая работа 3.3. Множественное ветвление

Практическая работа 3.4. Условные операторы

### Раздел 4. Циклы (10 часов)

Теория: Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randint. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. Числа Фибоначчи

Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов

Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом

### Раздел 5. Строки - последовательности символов (4 часа)

Теория: Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк.Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки,списки,кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2.Решение задач со строками.

### **Условия реализации программы**

#### ***Материально-техническоеобеспечение:***

- столы для компьютера;
- компьютерные стулья;
- шкафы для дидактических материалов, пособий;
- канцтовары;

#### ***Информационное обеспечение:***

- персональный компьютер (на каждого участника);
- мультимедийный проектор;
- оргтехника;
- выход в сеть Internet;

#### ***Аппаратное обеспечение:***

- Процессор не нижеCore2Duo;
- Объем оперативной памяти не ниже 4 Гб DDR3;
- Дисковое пространство не менее 128 Гб;
- Монитор диагональю не менее 18’;

#### ***Программное обеспечение:***

- Операционная система Windows7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- IDE Jat Brains PyCharm;
- Foxit Reader или другой просмотрщик PDF файлов;
- WinRAR;
- Пакетофисных программ;
- AdobePhotoshop или другой растровый графическийредактор;
- Любой браузер для интернет серфинга.

### **Характеристика помещений для занятий по программе**

Теоретические занятия проходят в компьютерном классе, который соответствует нормам СанПиН, оборудованном современными ПК с ЖК-мониторами, мультимедийным проектором, стендами с литературой.

#### **Информационное обеспечение:**

- видеоматериалы разной тематики по программе;
- цифровые разработки педагога к занятиям (презентации, сайты, тесты и т.д.)

### **Определение результативности**

#### **Формы аттестации (контроля)**

**Входящий контроль** осуществляется при комплектовании группы в начале учебного года. *Цель* – определить исходный уровень знаний учащихся, определить формы и методы работы с учащимися.

*Форма контроля:* тестирование.

**Текущий контроль** осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу учащихся, позволяют более полно проявлять полученные знания, умения, навыки.

**Промежуточный контроль** осуществляется в конце I полугодия учебного года.

*Форма контроля:* тест, решение практических задач средствами языка программирования.

**Итоговый контроль** осуществляется в конце учебного года. *Форма контроля:* защита творческого проекта.

Отслеживание личностного развития учащихся осуществляется методом наблюдения, анкетирования. По итогам первого полугодия и по итогам года заполняется «Диагностическая карта», в которой проставляется уровень усвоения программы каждым учащимся объединения.

### **Краткое описание методики работы по программе**

**Особенности организации образовательного процесса** – очная форма.

#### **Методы обучения:**

1. **Словесный:** объяснение нового материала; рассказ обзорный для раскрытия новой темы; беседы с учащимися в процессе изучения темы.
2. **Наглядный:** применение демонстрационного материала, наглядных пособий, презентаций по теме.
3. **Практический:** индивидуальная и совместная продуктивная деятельность, выполнение учащимися определенных заданий, решение задач.
4. **Интерактивный:** создание специальных заданий, моделирующих реальную жизненную ситуацию, из которой учащимся предлагается найти выход.

#### **Формы организации образовательного процесса**

Индивидуально-групповая и групповая

#### **Формы организации учебного занятия**

Отбор методов обучения обусловлен необходимостью формировать информационную и коммуникативную компетентности учащихся, реализовывать личностно-ориентированное обучение, направлять их на самостоятельное решение разнообразных проблем, развивать исследовательские и творческие способности.

Решение данных задач кроется в организации деятельностного подхода к обучению, в проблемном изложении материала педагогом, в переходе от репродуктивного вида работ к самостоятельным, поисково-исследовательским



видам деятельности.

Поэтому основная методическая установка в данном курсе – обучение учащихся навыкам самостоятельной творческой деятельности.

Теория преподносится в форме беседы, эвристической беседы, презентации, обзора и т.п.

Практические занятия проходят в форме тестирования, практикума, совместной продуктивной деятельности, моделирования и защиты проектов, проблемном изложении материала, с помощью которого дети сами решают познавательные задачи.

### **Педагогические технологии**

1. Технология проблемного диалога. Учащимся не только сообщаются готовые знания, но и организуется такая их деятельность, в процессе которой они сами делают «открытия», узнают что-то новое и используют полученные знания и умения для решения жизненных задач.

2. Технология коллективного взаимообучения («организованный диалог», «сочетательный диалог», «коллективный способ обучения (КСО), «работа учащихся в парах сменного состава») позволяет плодотворно развивать у обучаемых самостоятельность и коммуникативные умения.

3. Игровая технология. Игровая форма в образовательном процессе создаётся при помощи игровых приёмов и ситуаций, выступающих как средство побуждения к деятельности. Способствует развитию творческих способностей, продуктивному сотрудничеству с другими учащимися. Приучает к коллективным действиям, принятию решений, учит руководить и подчиняться, стимулирует практические навыки, развивает воображение.

4. Элементы здоровьесберегающих технологий являются необходимым условием снижения утомляемости и перегрузки учащихся.

5. Проектная технология предлагает практические творческие задания, требующие от учащихся их применение для решения проблемных заданий, знания материала на данный исторический этап. Овладевая культурой проектирования, школьник приучается творчески мыслить, прогнозировать возможные варианты решения стоящих перед ним задач.

6. Информационно-коммуникационные технологии активизируют творческий потенциал учащихся; способствует развитию логики, внимания, речи, повышению качества знаний; формированию умения пользоваться информацией, выбирать из нее необходимое для принятия решения, работать с разными видами информации, программным обеспечением, специальными программами и т.д.

### **Алгоритм учебного занятия**

Теоретические занятия строятся следующим образом:

- заполняется журнал присутствующих на занятиях обучаемых;
- объявляется тема занятий;
- раздаются материалы для самостоятельной работы и повторения материала или указывается где можно взять этот материал;
- теоретический материал преподаватель дает обучаемым, помимо вербального, классического метода преподавания, при помощи различных современных технологий в образовании (аудио, видео лекции, экранные видео

лекции, презентации, интернет, электронные учебники);

- проверка полученных знаний осуществляется при помощи тестирования обучаемых.

Практические занятия проводятся следующим образом:

- преподаватель показывает конечный результат занятия, т.е. заранее готовит практическую работу;

- далее преподаватель показывает, используя различные варианты, последовательность работы;

- преподаватель отдает обучаемым, ранее подготовленные самостоятельно мультимедийные материалы по изучаемой теме, либо показывает где они размещены на его сайте посвященном именно этой теме;

- далее обучаемые самостоятельно(и, или) в группах разрабатывают проект;

- практические занятия начинаются с правил техники безопасности при работе с различным инструментом и с электричеством и разбора допущенных ошибок во время занятия в обязательном порядке.

## Дидактические материалы

- специальная и научно-популярная литература для педагога и учащихся;
- раздаточные материалы, инструкционные, технологические карты, задания, упражнения, образцы изделий и т.п.

### Список литературы для педагога

1. Бизли, Дэвид М. Python. Подробный справочник. – М. – СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник.–М.: Вильямс, 2015.
3. Официальный сайт программы[Электронный ресурс] –Режим доступа: <https://docs.python.org/>, свободный.
4. Сайт, среда разработки для языка Python.[Электронный ресурс]–Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.

### Список литературы для родителей и учащихся

1. Сайт/справочные материалы [Электронный ресурс] – Режим доступа: <https://metanit.com/python/>, свободный.
2. Сайт/интерактивный сборник задач для практики программирования [Электронный ресурс]– Режим доступа: <http://pythontutor.ru/>, свободный.
3. Сайт/Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: <https://stepik.org/course/431>, свободный.
4. Сайт/среда разработки для языка Python [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.
5. Сайт проекта OpenBookProject. Практические примеры на Python Криса Мейерса [Электронный ресурс] – Режим доступа: [openbookproject.net](http://openbookproject.net), свободный.