

Управление образования администрации округа Муром
Муниципальное бюджетное общеобразовательное учреждение
«Гимназия №6»

ПРИНЯТО:

на заседании педагогического совета
Протокол № 8 от 31.05, 2023г



УТВЕРЖДЕНО:

и.о. директора МБОУ «Гимназия №6»
О.А. Кожемякина

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ (ОБЩЕРАЗВИВАЮЩАЯ) ПРОГРАММА

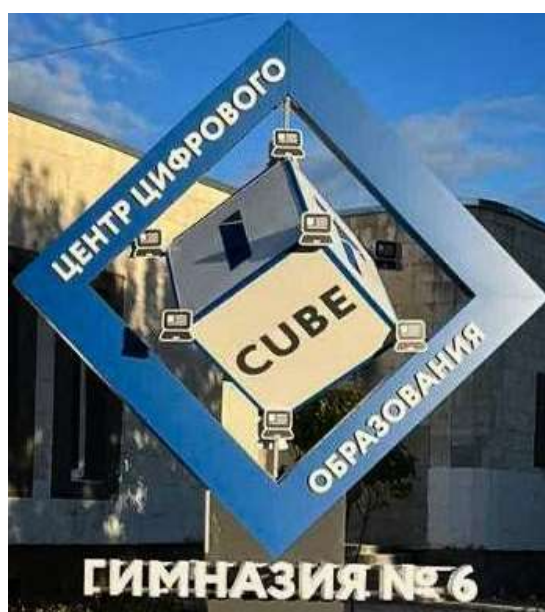
«Программирование на языке Python»

Продвинутый уровень

Направленность – техническая

Категория обучающихся: 13 – 17 лет

Срок реализации программы – 1 год (76 часов)



2023г, Муром



1. Комплекс основных характеристик программы

1.1 Пояснительная записка

Программа «Программирование на Python» (продвинутый уровень) рассчитана на обучающихся 13–17 лет, а также имеет техническую направленность. Целью программы является формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python. Данная образовательная программа интегрирует в себе достижения сразу нескольких традиционных направлений, а главной её особенностью является большой блок практических заданий, и самостоятельная работа над решением поставленных задач: обучающиеся учатся решать задачи без помощи преподавателя, что способствует развитию у них навыков алгоритмического и логического мышления, умению мыслить самостоятельно и повышает мотивацию к обучению. В ходе занятий обучающиеся приобретают знания и умения, которые могут быть использованы ими при дальнейшей сдаче ЕГЭ, при участии в олимпиадах по программированию, при решении задач по физике, химии, биологии, лингвистике и другим наукам, а также они являются фундаментом для дальнейшего совершенствования мастерства программирования.

Циклы, ветвления и функции – все это элементы структурного программирования, возможностей которого вполне хватает для написания небольших, простых программ и сценариев. Однако крупные проекты часто реализуют, используя парадигму объектно-ориентированного программирования (ООП). Python позволяет познакомиться с обеими парадигмами, изучить их различие.

Дополнительная общеразвивающая программа «Программирование на Python» (продвинутый уровень) реализуется посредством интеграции с предметом «Информатика» в 10-11 классах, а также в рамках внеурочной и кружковой деятельности. Основанием для проектирования и реализации данной дополнительной общеразвивающей образовательной программы служит *перечень следующих нормативных правовых актов и государственных программных документов:*

– Конституция Российской Федерации (принята всенародным голосованием 12.12.1993 с изменениями, одобренными в ходе общероссийского голосования 01.07.2020). — URL: http://www.consultant.ru/document/cons_doc_LAW_28399/

– Федеральный закон от 29.12.2012 № 273-ФЗ (ред. от 31.07.2020) «Об образовании в Российской Федерации» (с изм. и доп., вступ. в силу с 01.09.2020). — URL: http://www.consultant.ru/document/cons_doc_LAW_140174/

– Паспорт национального проекта «Образование» (утверждён президиумом Совета при Президенте РФ по стратегическому развитию и национальным проектам, протокол от 24.12.2018 № 16). — URL: http://www.consultant.ru/document/cons_doc_LAW_319308/

– Государственная программа Российской Федерации «Развитие образования» (утверждена постановлением Правительства РФ от 26.12.2017 № 1642 (ред. от 22.02.2021) «Об утверждении государственной программы Российской Федерации «Развитие образования»). — URL: http://www.consultant.ru/document/cons_doc_LAW_286474/

– Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства РФ от 29.05.2015 № 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»). — URL: http://www.consultant.ru/document/cons_doc_LAW_180402/

– Профессиональный стандарт «Педагог (педагогическая деятельность в дошкольном, начальном общем, основном общем, среднем общем образовании), (воспитатель, учитель)» (ред. от

16.06.2019) (приказ Министерства труда и социальной защиты РФ от 18 октября 2013 г. № 544н, с изменениями, внесёнными приказом Министерства труда и соцзащиты РФ от 25 декабря 2014 г. № 1115н и от 5 августа 2016 г. № 422н). — URL: http://www.consultant.ru/document/cons_doc_LAW_155553/

–Профессиональный стандарт «Педагог дополнительного образования детей и взрослых» (приказ Министерства труда и социальной защиты РФ от 5 мая 2018 г. № 298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»). — URL: https://profstandart.rosmintrud.ru/obshchiy-informatsionnyy-blok/natsionalnyy-reestr-professionalnykh-standartov/reestr-professionalnykh-standartov/index.php?ELEMENT_ID=48583

–Федеральный государственный образовательный стандарт основного общего образования (утверждён приказом Министерства образования и науки Российской Федерации от 17 декабря 2010 г. № 1897) (ред. 21.12.2020). — URL: <https://fgos.ru>

–Федеральный государственный образовательный стандарт среднего общего образования (утверждён приказом Министерства образования и науки Российской Федерации от 17 мая 2012 г. № 413) (ред. 11.12.2020). — URL: <https://fgos.ru>

–Методические рекомендации по созданию и функционированию детских технопарков «Кванториум» на базе общеобразовательных организаций (утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р-4). — URL: http://www.consultant.ru/document/cons_doc_LAW_374695/

–Методические рекомендации по созданию и функционированию центров цифрового образования «ИТ-куб» (утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р-5). — URL: http://www.consultant.ru/document/cons_doc_LAW_374572/

–Методические рекомендации по созданию и функционированию в общеобразовательных организациях, расположенных в сельской местности и малых городах, центров образования естественно-научной и технологической направленностей («Точка роста») (утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р-6). — URL: http://www.consultant.ru/document/cons_doc_LAW_374694/

Актуальность программы. Научившись программировать на языке Python, учащиеся получают мощный и удобный инструмент для решения как учебных, так и прикладных задач. В том числе рассматриваются различные варианты решения задач ЕГЭ, а также решение задач различных дисциплин (математики, физики, химии, биологии) с применением программирования.

Также на основе полученных представлений о программировании можно с легкостью выучить и другие языки.

В основу курса заложены принципы модульности и практической направленности. Содержание учебных модулей направлено на:

- детальное изучение алгоритмизации;
- реализацию межпредметных связей;
- организацию проектной и исследовательской деятельности обучающихся.

Кроссплатформенность языка и практика создания объектно-ориентированных программ, а также изучение особенностей создания графического интерфейса, позволяют получить наиболее полное представление о деятельности современного программиста и стоящих перед ним задач. Это имеет ценность при выборе будущей профессии.

Направленность программы. Программа обладает технической направленностью и предназначена для использования в системе дополнительного образования детей. Обучающиеся в

ходе занятий приобщаются к инженерно-техническим знаниям в области информационных технологий, формируют логическое мышление.

Адресат программы. Дополнительная программа «Программирование на Python» (продвинутый уровень) предназначена для детей в возрасте 13 – 17 лет, проявляющих интерес к программированию, без ограничений возможности здоровья.

Содержание программы составлено с учётом возрастных и психологических особенностей детей данного возраста.

Формы проведения занятий. По форме организации деятельности предполагаются групповые занятия, количество обучающихся в группе – 8-12 человек.

Место проведения занятий: г. Муром, ул. Комсомольская, д. 60.

Режим занятий. Занятия проводятся 1 раз в неделю по 2 академических часа с перерывом 10 минут для проветривания помещения и отдыха обучающихся. Продолжительность одного занятия – 40 минут.

Условия реализации программы. Требуется предварительного освоения программы базового уровня по направлению «Основы программирования на Python».

Срок освоения общеразвивающей программы определяется содержанием программы и составляет 5 месяцев (76 часов).

Формы обучения: очная, с применением дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Виды занятий: беседы, обсуждения, игровые формы работы, практические занятия, метод проектов. Также программа курса включает групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия).

По типу организации взаимодействия педагога с обучающимися при реализации программы используются личностно-ориентированные технологии, технологии сотрудничества.

Здоровьесберегающая деятельность реализуется:

- созданием безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности обучающихся;
- контролем педагога за соблюдением обучающимися правил работы за персональным компьютером;
- созданием благоприятного психологического климата в учебной группе в целом.

Уровень освоения программы. Уровень освоения дополнительной общеразвивающей программы – *продвинутый*, требующий предварительного освоения программы базового уровня.

Цель программы заключается в изучении методов программирования на языке Python, объектно-ориентированной парадигмы программирования, её сравнение со структурной, а также в создании графических интерфейсов, подготовке к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

Для успешной реализации поставленной цели необходимо решить ряд поставленных задач:

Обучающие (предметные):

- сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ;
- ознакомить с принципами и методами объектно-ориентированного программирования;
- сформировать навыки работы в интегрированной среде разработки на языке Python;
- изучить конструкции языка программирования Python;



–ознакомить с основными понятиями объектно-ориентированного программирования:
наследование, полиморфизм, инкапсуляция;

–сформировать навыки разработки графического интерфейса программ;

–сформировать навыки разработки эффективных алгоритмов и программ на основе изучения
языка программирования Python;

–сформировать понимание оптимального решения;

–сформировать навыки поиска информации в сети Интернет, анализ выбранной информации
на соответствие запросу, использование информации при решении задач.

Развивающие (метапредметные):

–развить умение самостоятельно ставить и формулировать для себя новые задачи, развивать
мотивы своей познавательной деятельности;

–развить умение самостоятельно планировать пути решения поставленной проблемы для
получения эффективного результата; понимание, что в программировании длинная программа не
всегда лучшая, выбрать оптимальный способ решения задачи;

–развить умение критически оценивать правильность решения учебно-исследовательской
задачи;

–развить умение корректировать свои действия, вносить изменения в программу и отлаживать
её в соответствии с изменяющимися условиями;

–сформировать владение основами самоконтроля, способность к принятию решений;

–развить умение создавать, применять и преобразовывать знаки и символы, модели и схемы
для решения учебно-исследовательских и проектных работ;

–сформировать компетентность в области использования информационно-
коммуникационных технологий (ИКТ-компетенция);

–развить умение организовывать учебное сотрудничество и совместную деятельность с
учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Воспитательные (личностные):

–сформировать ответственное отношение к учению, способности довести до конца начатое
дело аналогично завершённым творческим учебным проектам;

–сформировать способности к саморазвитию и самообразованию средствами
информационных технологий на основе приобретённой мотивации к обучению и познанию;

–развить опыт участия в социально значимых проектах, повышение уровня самооценки
благодаря реализованным проектам;

–сформировать коммуникативную компетенцию в общении и сотрудничестве со
сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности,
участия в конкурсах и олимпиадах различного уровня;

–сформировать целостное мировоззрение, соответствующее современному уровню развития
информационных технологий;

–сформировать осознанное позитивное отношение к другому человеку, его мнению,
результату его деятельности;

–сформировать ценность здорового и безопасного образа жизни;

–обеспечить усвоение правил индивидуального и коллективного безопасного поведения при
работе с компьютерной техникой.

Обучающийся должен уметь:

–объяснять и использовать на практике как простые, так и сложные структуры данных и
конструкций для работы с ними;

–искать и обрабатывать ошибки в коде;



- для решения задач делить их на подзадачи;
- писать грамотный красивый код;
- находить, оценивать, использовать информацию из различных источников, необходимую для решения профессиональных задач, в том числе на основе системного подхода;
- грамотно строить коммуникацию, исходя из целей и ситуации.

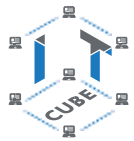
Обучающийся должен знать:

- основы современных языков программирования;
- практическую направленность с ориентацией на реальные потребности, соответствующие возрасту обучающихся.

1.2 Содержание дополнительной общеразвивающей программы

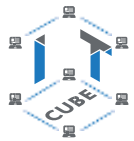
1.2.1 Содержание учебного плана

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
РАЗДЕЛ 1: Объектно-ориентированное программирование					
1.1	Что такое ООП	2	1	1	Беседа
1.2	Создание классов и объектов	6	2	4	Практическая работа
1.3	Наследование	4	1	3	Тестирование
1.4	Полиморфизм	4	1	3	Тестирование
1.5	Инкапсуляция	6	2	4	Тестирование
1.6	Перезагрузка операторов	4	1	3	Тестирование
1.7	Разработка объектно-ориентированной программы.	10		10	Проект
РАЗДЕЛ 2: Графический интерфейс					
2.1	Примеры и возможности графических модулей в Python	4	1	3	Практическая работа
2.2	Модуль Tkinter	2	2		Тестирование
2.3	Виджеты	10	4	6	Тестирование
2.4	Холст	6	2	4	Тестирование
2.5	Методы pack, grid, place	4	1	3	Тестирование
2.6	Обработка событий	4	2	2	Тестирование
2.7	Создание приложения с графическим интерфейсом	8		8	Проект
2.8	Итоговая аттестация	2		2	Защита проекта
Итого		76	20	56	

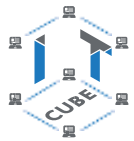


1.2.2 Тематическое планирование

№ п/п	Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, проект	Объём часов	Основные виды деятельности обучающихся	Использование оборудования
<i>Раздел 1. Объектно-ориентированное программирование</i>					
1	Тема 1. Что такое ООП	Содержание учебного материала	2	Наблюдение за работой учителя, выполнение базовых инструкций совместно с учителем	Ноутбук, IDLE
		Объектный подход в программировании. Основные понятия ООП: класс, экземпляр, атрибуты, методы.			
		Практическая работа №1. Моделирование класса			
2	Тема 2. Создание классов и объектов.	Содержание учебного материала	6	Создание собственного класса.	Ноутбук, IDLE
		Конструктор класса. Метод <code>__init__()</code> . Создание полей и методов класса. Создание экземпляров класса. Изменение атрибутов, вызов методов.			
		Практическая работа №2. Создание класса.			
3	Тема 3. Наследование.	Содержание учебного материала	4	Создание подклассов уже созданного класса, настройка их методов.	Ноутбук, IDLE
		Основные понятия ООП: наследование – сущность, назначение, применение. Иерархия классов. Создание классов-наследников. Расширение методов, переопределение методов.			
		Практическая работы №3. Создание подклассов.			
4	Тема 4. Полиморфизм.	Содержание учебного материала	4	Работа с классом.	Ноутбук, IDLE
		Основные понятия ООП: полиморфизм – сущность, назначение, применение.			
		Практические работы №4,5. Использование методов различных классов.			
5	Тема 5. Инкапсуляция.	Содержание учебного материала	6	Работа с классом.	Ноутбук, IDLE
		Основные понятия ООП: инкапсуляция – сущность, назначение, применение. Скрытие внутреннего устройства.			
		Практическая работы № 6. Скрытие методов и атрибутов класса.			



6	Тема 6. Перезагрузка операторов.	Содержание учебного материала	4	Анализ имеющегося функционала, поиск недостаточных ресурсов. Доработка классов.	Ноутбук, IDLE
		Перезагрузка операторов. Методы: <code>__del__()</code> , <code>__str__()</code> , <code>__add__()</code> , <code>__setattr__()</code>			
		Практические работы №7, 8, 9. Перезагрузка оператора сложения, вывода на экран, присваивания.			
7	Тема 7. Разработка объектно-ориентированной программы.	Содержание учебного материала	10	Разработка собственной программы.	Ноутбук, IDLE
		Моделирование компьютерной игры. Создание персонажей: описание атрибутов и методов. Построение иерархии классов в программе. Настройка взаимодействия методов.			
		Разработка игры.			
<i>Раздел 2. Графический интерфейс</i>					
8	Тема 1. Примеры и возможности графических модулей в Python	Содержание учебного материала	4	Наблюдение за работой учителя, выполнение базовых инструкций совместно с учителем	Ноутбук, IDLE
		Графический интерфейс. Элементы графического интерфейса. Модуль Turtle, Tkinter – сравнение, возможности.			
		Практическая работа №1. Черепашья графика.			
9	Тема 2. Модуль Tkinter	Содержание учебного материала	2	Наблюдение за работой учителя, выполнение базовых инструкций совместно с учителем	Ноутбук, IDLE
10	Тема 3. Виджеты	Содержание учебного материала	10	Написание программ.	Ноутбук, IDLE
		Виджеты Button, Label, Entry. Размещение кнопок и окон ввода в окне. Ввод и вывод данных программы через графическое окно. Создание приложений.			
		Практические работы №2,3,4. Написание калькулятора, игры крестики-нолики и змейка.			
11	Тема 4. Холст	Содержание учебного материала	6	Рисование статических и анимированных объектов.	Ноутбук, IDLE
		Использование холста. Canvas. Размещение примитивов. Идентификаторы, теги, анимация.			
		Практические работы № 5,6. Рисование объектов.			



12	Тема 5. Методы pack, grid, place	Содержание учебного материала	4	Написание программы.	Ноутбук, IDLE
		Расположение объектов в окне программы. Координаты, сетка.			
		Практическая работа №7. Разработка интерфейса программы.			
13	Тема 6. Обработка событий	Содержание учебного материала	4	Написание программы.	Ноутбук, IDLE
		Реакция на нажатие кнопок клавиатуры и мыши.			
		Практическая работа №8. Создание графического редактора.			
14	Тема 7. Создание приложения с графическим интерфейсом	Содержание учебного материала	8	Разработка собственной программы.	
		Графическое оформление своей игры.			
		Разработка игры.			
15	Итоговая аттестация	Презентация созданного приложения.	2	Защита проекта	Ноутбук, IDLE
ИТОГО:			76		



1.2.3 Содержание занятий

Модуль 1. Объектно-ориентированное программирование

Тема 1.1 Что такое ООП

Теория: Объектный подход в программировании. Основные понятия ООП: класс, экземпляр, атрибуты, методы.

Практика: Черепашня графика

Тема 1.2 Создание классов и объектов

Теория: Конструктор класса. Метод `__init__()`. Деструкторы. Создание полей и методов класса. Создание экземпляров класса. Изменение атрибутов, вызов методов. *Практика:* Создание класса (описание полей и функций)

Тема 1.3 Наследование

Теория: Основные понятия ООП: наследование – сущность, назначение, применение. Иерархия классов. Создание классов-наследников. Расширение методов, переопределение методов.

Практика: Создание подклассов

Тема 1.4 Полиморфизм

Теория: Основные понятия ООП: полиморфизм – сущность, назначение, применение.

Практика: Использование методов различных классов

Тема 1.5 Инкапсуляция

Теория: Основные понятия ООП: инкапсуляция – сущность, назначение, применение. Скрытие внутреннего устройства.

Практика: Скрытие внутреннего устройства класса.

Тема 1.6 Перегрузка операторов

Теория: Перегрузка операторов. Методы: `__del__()`, `__str__()`, `__add__()`, `__setattr__()`

Практика: Перегрузка операторов сложения, вывода на экран, присваивания.

Тема 1.7 Создание объектно-ориентированной программы

Теория: Моделирование компьютерной игры. Создание персонажей: описание атрибутов и методов. Построение иерархии классов в программе. Настройка взаимодействия методов.

Практика: Создание подклассов

Модуль 2. Объектно-ориентированное программирование

Тема 2.1 Примеры и возможности графических модулей в Python

Теория: Графический интерфейс. Элементы графического интерфейса. Модуль `Turtle`, `Tkinter` – сравнение, возможности.

Практика: Создание подклассов

Тема 2.2 Модуль Tkinter

Теория: Назначение модуля. Обзор функционала. Настройки графического окна.

Тема 2.3 Виджеты

Теория: Виджеты `Button`, `Label`, `Entry`. Размещение кнопок и окон ввода в окне. Ввод и вывод данных программы через графическое окно. Создание приложений.

Практика: Создание калькулятора, игры крестики-нолики и змейка.

Тема 2.4 Холст

Теория: Использование холста. `Canvas`. Размещение примитивов. Идентификаторы, теги, анимация.

Практика: Рисование объектов. **Тема 2.5 Методы `pack`, `grid`, `place`** *Теория:* Расположение объектов в окне программы. Координаты, сетка.



Практика: Разработка интерфейса программы.

Тема 2.6 Обработка событий

Теория: Реакция на нажатие кнопок клавиатуры и мыши.

Практика: Создание графического редактора.

Тема 2.7 Создание программы с графическим интерфейсом *Теория:* Графическое оформление своей игры *Практика:* Разработка игры.

Тема 2.8 Итоговая аттестация

Практика: Презентация созданной программы.

1.3 Планируемые результаты

Обучающие (предметные):

–сформированы и развиты навыки алгоритмического и логического мышления, грамотной разработки программ;

–ознакомлены с принципами и методами объектно-ориентированного программирования;

–сформированы навыки работы в интегрированной среде разработки на языке Python;

–изучены конструкции языка программирования Python;

–ознакомлены с основными структурами данных и типовыми методами обработки этих структур;

–сформированы навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

–сформированы навыки поиска информации в сети Интернет, умеет анализировать выбранную информацию на соответствие запросу, использует информацию при решении задач.

Развивающие (метапредметные):

–развито умение самостоятельно ставить и формулировать для себя новые задачи, развиты мотивы своей познавательной деятельности;

–развито умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата;

–развито умение критически оценивать правильность решения учебно- исследовательской задачи;

–развито умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;

–сформированы владение основами самоконтроля и способность к принятию решений;

–развито умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;

–сформирована компетентность в области использования информационно-коммуникационных технологий (ИКТ-компетенция);

–развито умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно- исследовательской деятельности.

Воспитательные (личностные):

–сформировано ответственное отношение к учению, способность довести до конца начатое дело аналогично завершённым творческим учебным проектам;

–сформированы способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой мотивации к обучению и познанию;



–развит опыт участия в социально значимых проектах, повысился уровень самооценки благодаря реализованным проектам;

–сформирована коммуникативная компетенция в общении и сотрудничестве со сверстниками в процессе образовательной, учебно- исследовательской и проектной деятельности, участвуют в конкурсах и конференциях различного уровня;

–сформировано целостное мировоззрение, соответствующее современному уровню развития информационных технологий;

–сформировано осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности;

–сформирована ценность здорового и безопасного образа жизни;

–обеспечено усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Обучающийся должен уметь:

–объяснять и использовать на практике как простые, так и сложные структуры данных и конструкций для работы с ними;

–искать и обрабатывать ошибки в коде;

–разбирать решение задач на подзадачи;

–писать грамотный красивый код;

–находить, оценивать, использовать информацию из различных источников, необходимую для решения профессиональных задач, в том числе на основе системного подхода;

–грамотно строить коммуникацию, исходя из целей и ситуации. *Обучающийся должен знать:*

–основы современных языков программирования;

–практическую направленность с ориентацией на реальные потребности, соответствующие возрасту обучающихся.

1.4 Методические материалы

Образовательный процесс осуществляется в очной форме.

В образовательном процессе используются следующие методы:

1. объяснительно-иллюстративный;

2. метод проблемного изложения (постановка проблемы и решение её самостоятельно или группой);

3. проектно-исследовательский;

- демонстрация плакатов, схем, таблиц, диаграмм;

- использование технических средств;

- просмотр кино- и телепрограмм;

4. практический:

- практические задания;

- анализ и решение проблемных ситуаций и т. д.

Выбор методов обучения осуществляется исходя из анализа уровня готовности обучающихся к освоению содержания модуля, степени сложности материала, типа учебного занятия. Вместе с объяснительно-иллюстративным используются, а также становятся ведущими частично-поисковый и проектно-исследовательский методы, проблемного изложения и проектно-исследовательский. На выбор методов обучения значительно влияет персональный состав группы, индивидуальные особенности, возможности и запросы детей.



Список литературы

Основная литература

1. Задачи по программированию. Под ред. С.М. Окулова. М.: БИНОМ, Лаборатория знаний, 2006.
2. Лутц, М. С. Изучаем PYTHON. СПб.: Символ-Плюс, 2011.
3. Окулов, С Основы программирования. М.: БИНОМ. Лаборатория знаний, 2012
4. Поляков К.Ю., Е.А. Еремин. Информатика. Углубленный уровень. Учебник для 10 класса в двух частях. М.: БИНОМ. Лаборатория знаний, 2014

Дополнительная литература

1. Информатика и ИКТ. Задачник – практикум в двух частях. Под ред. И.Г. Семакина и Е.К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.
2. Лутц, М. Изучаем PYTHON. СПб.: Символ-Плюс, 2011.

Интернет-ресурсы

1. «Python 3 для начинающих. [Электронный ресурс]: <https://pythonworld.ru>
2. «Питонтьютор». [Электронный ресурс]: <https://pythontutor.ru>
3. Материалы и презентации к урокам в LMS Яндекс. Лицея.