

Государственное бюджетное нетиповое образовательное учреждение

**САНКТ-ПЕТЕРБУРГСКИЙ ГОРОДСКОЙ ЦЕНТР
ДЕТСКОГО ТЕХНИЧЕСКОГО ТВОРЧЕСТВА**

РАССМОТРЕНО
на педагогическом совете
СПбГЦДТТ
Протокол № 1 от 30 августа 2023 г.

УТВЕРЖДЕНА
Приказом № 71/2 от 31.08.2023
Директор СПбГЦДТТ

_____ А.Н. Думанский

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

**«КРЕАТИВНОЕ ПРОГРАММИРОВАНИЕ
(Scratch+Python)»**

Возрастной состав обучающихся: 8 - 12 лет

Продолжительность обучения: 2 года

Разработчик:
Румянцева Мария Юрьевна,
педагог дополнительного образования
СПбГЦДТТ

Программа разработана: 2017 г.
Последняя корректировка: 2023 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеразвивающая программа «Креативное программирование (Scrath+Python)» реализуется в рамках деятельности отдела компьютерных технологий.

Направленность программы: техническая.

Программирование — процесс создания компьютерных программ.

По известному выражению Никлауса Вирта «Программы = алгоритмы + структуры данных»; иными словами, ключевыми непосредственными задачами программирования являются создание и использование алгоритмов и структур данных.

В более широком смысле под программированием понимают весь спектр деятельности, связанный с созданием и поддержанием в рабочем состоянии программ — программного обеспечения. Эта инженерно-техническая дисциплина называется «программная инженерия». Сюда входят анализ и постановка задачи, проектирование программы, построение алгоритмов, разработка структур данных, написание текстов программ, отладка и тестирование программы (испытания программы), документирование, настройка (конфигурирование), доработка и сопровождение.

Программирование основывается на использовании языков программирования, на которых записываются инструкции для компьютера. Современное приложение содержит множество таких инструкций, связанных между собой.

Программирование занимает одно из первых мест среди современных профессий по требованиям к владению навыками алгоритмирования и абстрагирования, а так же базовых навыков программирования, которыми необходимо овладеть как можно раньше из-за динамичности развития области и требовательности ее к навыкам адаптации к новым технологиям.

Актуальность программы обусловлена:

- потребностью в алгоритмически грамотных специалистах,
- необходимостью алгоритмической грамотности при изучении любого языка программирования,
- необходимостью профессионального самоопределения, предоставлением возможности определиться в дальнейшем развитии,
- быстрорастущими требованиями к новым поколениям программистов.

Адресат программы. Дети в возрасте 8-12 лет, интересующиеся программированием.

Уровень освоения программы: общекультурный

Цель программы: сформировать у учащихся системы компетенций, связанных со знаниями в области алгоритмизации и программирования, общих закономерностей развития IT-области для дальнейшего их применения при освоении различных областей программирования и выполнения работ, связанных с программированием и алгоритмизацией, включая проектные. Развить индивидуальные познавательные способности каждого ребенка; максимально выявить, инициировать, использовать, «окультурить» индивидуальный (субъектный) опыт ребенка; помочь личности познать себя, самоопределиться и самореализоваться, а не формировать заранее заданные свойства.

Задачи:

Обучающие:

- овладение учениками основными способами представления алгоритма;
- изучение основных элементарных алгоритмов на понятийном уровне;

- овладение способностью к анализу алгоритмов, встречающихся в повседневности;
- овладение навыками оптимизации «повседневных» алгоритмов;
- овладение способностью к созданию и анализу алгоритмов в среде «Scratch»;
- изучение принципов структурного программирования;

Развивающие

- развитие у учащихся логического мышления
- развитие способностей анализа, обобщения
- развитие аналитических способностей

Воспитательные

- привлечение внимания обучающихся и их родителей к деятельности ГБНОУ СПбГЦДТТ;
- формирование творческого отношения к практической деятельности;
- формирование навыков взаимодействия в обществе;
- профориентация, профессиональная проба;
- выявление и поддержка одарённых детей;
- формирование позитивной инклюзивной среды;
- воспитание патриота и гражданина своей Родины;
- формирование интереса к ЗОЖ.

С целью создания условий для повышения гражданской ответственности за судьбу страны, воспитания гражданина, любящего свою Родину и семью, имеющего активную жизненную позицию, учащихся принимают участие в мероприятиях, направленных на воспитание гражданственности и патриотизма, проводимых в СПбГЦДТТ, а также на занятиях проводятся беседы по патриотическому воспитанию.

В данной программе на занятиях предусмотрены пятиминутки для профилактики коррупционных действий с целью формирования правового сознания и антикоррупционного мировоззрения учащихся. Учащиеся принимают участие в информационно-просветительских мероприятиях СПбГЦДТТ.

Отличительной особенностью данной программы является направленность образовательного процесса на формирование способностей к алгоритмическому мышлению, крайне необходимому для дальнейшего развития учащегося в области программирования, а так же позволяющая выделить одаренных детей на раннем этапе их развития и способствующая раннему самоопределению в профессиональной области. Программа практикоориентирована, реализует паттерн learning by doing.

Также особенностью программы является абстрагированность от конкретных сред и языков программирования, что позволит сохранять актуальность программы без значительных изменений для различных педагогов.

Процесс достижения цели и задач программы осуществляется в сотрудничестве учащихся, педагога дополнительного образования и родителей.

Педагогические основы программы

Программа основана на следующих педагогических принципах:

- Личностной ориентированности
- Природосообразности
- Практичности
- Системности
- Наглядности
- Профессиональной целесообразности
- Политехнизма

Личностно-ориентированное обучение - призвано раскрыть и использовать субъектный опыт ученика, помочь становлению личностно значимых способов познания путем организации целостной учебной (познавательной) деятельности.

Природосообразность - педагогический принцип, согласно которому педагог в своей деятельности руководствуется факторами естественного, природного развития ребенка, согласования организации всего педагогического процесса не только с возможностями самого ребенка на определенных этапах его развития, но и с природой, в которой он живет, ее изменениями.

Основой программы является:

- самостоятельность в составлении и решении практических задач,
- запас технических знаний,
- богатство понятий, способность комбинировать, рассуждать,
- наличие технических интересов.

УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ:

Реализация дополнительной общеразвивающей программы осуществляется в очном режиме или с использованием дистанционных технологий и электронных образовательных ресурсов.

Данная программа рассчитана на детей 8-12 лет.

Продолжительность обучения – 2 года

Занятия проводятся 2 раз в неделю по 2 академических часа.

Программа нацелена на обучение основам алгоритмического мышления и способов разработки алгоритмов. Процесс обучения строится с учётом знаний, умений и навыков, приобретаемых школьниками на уроках в соответствии с обязательным образовательным минимумом.

В детское объединение «Креативное программирование (Scratch + Python)», на первый и обучения принимаются все желающие, без предварительного отбора, на второй году обучение принимаются по результатам итогового контроля первого года и желающие по результатам собеседования.

Следующим этапом после окончания обучения по программе «Креативное программирование» является переход на программу «Проектная деятельность в сфере компьютерных технологий» или на программу «Алгоритмы программирования».

Нормативные документы

1. Федеральный закон Российской Федерации №273-ФЗ "Об образовании в Российской Федерации" от 29.12.2012
2. Федеральный закон Российской Федерации №304-ФЗ "О внесении изменений в Федеральный закон "Об образовании в Российской Федерации" по вопросам воспитания обучающихся" от 31.07.2020
3. Об утверждении плана мероприятий по реализации в 2021-2025 годах Стратегии развития воспитания в Российской Федерации на период до 2025 года // Распоряжение Правительства Российской Федерации от 12.11.2020 № 2945-Р
4. Об утверждении санитарных правил СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи" // Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 №28
5. Санитарно-эпидемиологические требования к устройству, содержанию и организации работы образовательных организаций и других объектов социальной инфраструктуры для детей и молодежи в условиях распространения новой коронавирусной инфекции (COVID-19) // Постановление Главного государственного санитарного врача Российской Федерации от 30.06.2020 №16
6. Об утверждении Порядка зачета организацией, осуществляющей образовательную деятельность, результатов освоения обучающимися учебных предметов, курсов, дисциплин (модулей), практики, дополнительных образовательных программ в других организациях, осуществляющих образовательную деятельность // Приказ Министерства науки и высшего образования Российской Федерации // Министерства просвещения Российской Федерации от 30.08.2020 № 845/369
7. Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ // Приказ Министерства науки и высшего образования Российской Федерации / Министерства просвещения Российской Федерации от 05.08.2020 № 882/391
8. Методические рекомендации для субъектов Российской Федерации по вопросам реализации основных и дополнительных общеобразовательных программ в сетевой форме // Утв. Министерством просвещения Российской Федерации от 28.06.2019 № МР-81/02вн
9. Концепция развития дополнительного образования детей в Российской Федерации до 2030 года // Распоряжение Правительства РФ от 31.03.2022 n 678-р
10. Стратегия развития воспитания в Российской Федерации на период до 2025 года // Распоряжение Правительства Российской Федерации от 29.05.2015 №996-р
11. Национальный проект "Образование" // Протокол от 03.09.2018 №10 Президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам
12. Об объявлении в Российской Федерации Десятилетия детства // Указ Президента Российской Федерации от 29.05.2017 №240
13. Об утверждении Правил выявления детей, проявивших выдающие способности, сопровождения и мониторинга их дальнейшего развития // Постановление Правительства Российской Федерации от 17.11.2015 №1239
14. Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам // Приказ Министерства просвещения Российской Федерации от 09.11.2018 № 196 (с изм. (с изменениями на 30.09.2020 года))
15. Порядок применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ // Приказ Министерства образования и науки Российской Федерации от 23.08.2017 №816
16. О патриотическом воспитании в Санкт-Петербурге // Закон Санкт-Петербурга от 18.07.2016 № 453-87

17. Об утверждении Плана мероприятий по реализации в 2020-2025 годах Стратегии развития воспитания в Российской Федерации на период до 2025 года // Распоряжение Правительства Санкт-Петербурга от 21.08.2020 №24-рп
18. О внедрении методологии (целевой модели) наставничества в государственных образовательных учреждениях, находящихся в ведении Комитета по образованию и администраций районов Санкт-Петербурга // Распоряжение Комитета по образованию Санкт-Петербурга от 27.07.2020 № 1457-р
19. Концепция воспитания юных петербуржцев на 2020-2025 годы "Петербургские перспективы" // Распоряжение Комитета по образованию Санкт-Петербурга от 16.01.2020 №105-р
20. Методические рекомендации по проектированию дополнительных общеразвивающих программ в государственных образовательных организациях Санкт-Петербурга, находящихся в ведении Комитета по образованию // Распоряжение Комитета по образованию Санкт-Петербурга от 01.03.2017 № 617-р
21. Методические рекомендации о механизмах привлечения организаций дополнительного образования детей к профилактике правонарушений несовершеннолетних // Письмо Комитета по образованию Санкт-Петербурга от 19.05.2017 № 03-12-274/17-0-2

Техническое оснащение занятий:

- Компьютерный класс, оборудованный персональными компьютерами.
- Интерактивная доска.
- Программное обеспечение:
 - Среда визуального программирования «Scratch».
 - Интерпретатор языка программирования Python.
 - Среда разработки PyCharm Community.
 - Современный браузер.

Программное обеспечение процесса обучения

Scratch (Скретч) — визуальная объектно-ориентированная среда программирования для обучения школьников младших и средних классов. Название произошло от слова *scratching* — техники, используемой хип-хоп-диджеями, которые крутят виниловые пластинки взад-вперёд руками для того, чтобы смешивать музыкальные темы.

Скретч создан как продолжение идей языка Лого и конструктора Лего. Скретч 1 был написан на языке Squeak, Скретч 2 ориентирован на работу онлайн и переписан на Flash/ActiveScript. Скретч разрабатывается небольшой командой программистов для детей в Массачусетском технологическом институте. В настоящий момент в курсе используется актуальная версия Scratch – Scratch 3, написанный на HTML 5.

Программы на Скретче состоят из графических блоков, подписи к которым зависят от выбранного для интерфейса языка. Может быть выбран один из 50 языков интерфейса, включая русский. Для подключения интерфейса на новом языке используются стандартные gettext файлы.

Python - высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ. Язык является полностью объектно-ориентированным в том плане, что всё является объектами. Необычной особенностью языка является выделение блоков кода пробельными отступами. Синтаксис ядра языка

минималистичен, за счёт чего на практике редко возникает необходимость обращаться к документации. Сам же язык известен как интерпретируемый и используется в том числе для написания скриптов. Язык из-за большого количества «синтаксического сахара» - условностей кода, облегчающих процесс написания программ, большого количества библиотек, нестрогой типизации подходит для начального обучения программирования. В курсе используется официальный интерпретатор языка Python.

Среда разработки PyCharm Community - открытая, доступная среда для разработки ПО на языке Python. Позволяет облегчить разработку и сборку программных продуктов с помощью проверки ошибок и подсказок.

Браузер – для использования в процессе курсы подходит любой современный браузер (основанные на движке Chromium или семейства Mozilla Firefox).

Все использующиеся программные продукты являются свободно распространяемыми.

В данной программе используется технология интерактивного обучения на основе использования новейших технических средств, состоящая из следующих этапов:

Первый год обучения:

- Изучение основных элементарных алгоритмов на понятийном уровне;
- Анализ алгоритмов, встречающихся в повседневности;
- Оптимизация примеров «повседневных» алгоритмов;
- Ввод понятия подпрограмма (функция)
- Создание алгоритмов в программной системе для изучения программирования на начальном уровне;
- Анализ созданных алгоритмов;
- Изучение принципов структурного программирования

Второй год обучения:

- Изучение основных конструкций языка программирования.
- Написание программ на базе языке программирования.
- Изучение теоретических основ программирования.
- Написание собственного проекта с использованием языка программирования.

Основными принципами используемой системы обучения являются:

- Принцип обратной связи. Используется как педагогом, так и учащимся. Учащимся обратная связь может быть нужна для понимания учебного материала, для коррекции приоритетов с учетом конкретных способностей и возможностей учащихся.
- Разделение материала программы на отдельные самостоятельные, но взаимосвязанные порции информации и учебные задания. Последовательность шаговых учебных процедур образует обучающую программу - основу технологии программированного обучения.
- Индивидуальность работы учащихся по программе. Каждый учащийся имеет возможность продвигаться в учении с наиболее благоприятной для его познавательных сил скоростью.
- Принцип использования специальных технических средств для подачи программируемых учебных материалов подразумевает использование компьютера для подачи некоторых частей программы и контроля их усвоения.

Программа логически разделена на 4 блока продолжительностью по полгода.

В первом блоке программа больше нацелена на развитие навыка разработки алгоритма с помощью выполнения заданий в программной системе для изучения базовых навыков программирования.

Во втором блоке программы рассматриваются более сложные алгоритмы, изучаются формальные способы представления алгоритмов.

В третьем и четвертом блоке работа ведется на базе языка программирования высокого уровня, и рассматриваются различные, постепенно усложняющиеся, алгоритмы, так же изучаются некоторые элементарные алгоритмы, имеющих особое значение в программировании, также особое внимание уделяется «красоте» написания кода.

В четвертом блоке программы уделяется особое внимание оптимизации алгоритмов.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ.

К концу обучения учащиеся должны овладеть следующими ключевыми компетенциями:

Личностные

- Культура работы с программным кодом
- Приобретение навыков командного решения задач

Предметные

- Знание основных терминов алгоритмизации и программирования
- Знание принципов процедурного программирования
- знание и понимание цели каждого из этапов создания программного продукта
- Знание основных принципов написания «Красивого» программного кода
- Знание основных принципов написания «Красивого» программного кода
- Знание этапов создания программного продукта
- Умение создать алгоритм любой сложности, основанный на принципах процедурного программирования
- Умение конкретизировать и абстрагировать алгоритм
- Умение работать в среде программирования «ПиктоМир», «КуМир»
- Умение использовать ПК для создания программных продуктов
- Умение представить алгоритм в виде псевкода
- Умение составить грамотное техническое задание к программному продукту
- Умение работать в команде
- Умение принимать социальные и этические обязательства

Метапредметные

- Развитие у учащихся логического мышления
- Развитие способностей анализа, обобщения
- Развитие аналитических способностей

Формы контроля

Контроль уровня усвоения знаний проводится по мере прохождения ключевых тем и разделов программы и включает в себя выполнения различного вида практических работ. Кроме этого, в течении каждого учебного года проводится вводный (контроль начального уровня подготовки детей), текущий (контроль уровня знаний проводящийся в середине каждого года обучения) и итоговый контроль (контроль степени усвоения программы определенного года обучения в целом).

УЧЕБНЫЙ ПЛАН ПЕРВОГО ГОДА ОБУЧЕНИЯ

№	Наименование раздела, темы	Кол-во часов			Формы контроля
		Всего	Теория	Практика	
1.	Вводное занятие. Задачи обучения. Техника безопасности. История информационных технологий.	2	1	1	Практическое задание.
2.	Понятие алгоритма.	2	1	1	Опрос
3.	Знакомство программной системой изучения программирования.	4	2	2	Тест
4.	Изучение линейных алгоритмов	12	6	6	Практическое задание
5.	Изучение условных алгоритмов	14	7	7	Практическое задание
6.	Изучение циклических алгоритмов	14	7	7	Практическое задание
7.	Изучение функций и подпрограмм	14	7	7	Практическое задание
8.	Повторение и закрепление пройденного материала	8	2	6	
9.	Итоговое занятие.	2	-	2	-
10.	Итого	160			

УЧЕБНЫЙ ПЛАН ВТОРОГО ГОДА ОБУЧЕНИЯ

	Наименование раздела, темы	Кол-во часов			Формы контроля
		Всего	Теория	Практика	
1.	Вводное занятие. Задачи обучения. Техника безопасности. История языков программирования.	2	1	1	-
2.	Изучение среды и языка программирования	20	10	10	Тест

3.	Структурное программирование: основы.	8	4	4	Практическое задание
4.	«Красивый» код.	6	3	3	Опрос
5.	Алгоритмы.	20	10	10	Практическое задание
6.	Разработка собственного проекта.	14	7	7	-
7.	Повторение и закрепление пройденного материала	8	2	6	
8.	Итоговое занятие.	2	-	2	-
	Итого	160			

Календарный учебный график

Год обучения	Дата начала обучения по программе	Дата окончания обучения по программе	Всего учебных недель	Количество учебных часов	Режим занятий
1 год	1 сентября	По мере выполнения программы 1 года обучения	40	80 часов по учебному расписанию	1 раз в неделю по 2 часа
2 год	1 сентября	По мере выполнения программы 2 года обучения	40	80 часов по учебному расписанию	1 раз в неделю по 2 часа

Утверждаю
Директор СПбГЦДТТ
_____ А.Н. Думанский

РАБОЧАЯ ПРОГРАММА

к дополнительной общеразвивающей программе

«КРЕАТИВНОЕ ПРОГРАММИРОВАНИЕ (Scratch+Python)»

20__ - 20__ учебный год

Год обучения 1

Группа № _____

Румянцева Мария Юрьевна,
педагог дополнительного
образования СПбГЦДТТ

Рабочая программа 1-го года обучения составлена на основе дополнительной общеразвивающей программы «Креативное программирование (scratch+python)» технической направленности.

Пояснительная записка

Цель первого года обучения: Сформировать у учащихся системы компетенций, связанных со знаниями в области алгоритмизации, общих закономерностей развития IT-области для дальнейшего их применения при освоении различных областей программирования и выполнения работ, связанных с программированием и алгоритмизацией.

Задачи 1 года обучения:

Обучающие:

- изучение основных элементарных алгоритмов на понятийном уровне;
- анализ алгоритмов, встречающихся в повседневности;
- оптимизация примеров «повседневных» алгоритмов;
- создание алгоритмов в среде «Scratch»;
- анализ созданных алгоритмов;
- изучение принципов структурного программирования;
- изучения способа представления алгоритма «Блок-схема»;
- реализация алгоритмов с использованием блок-схем.
- приобретение учениками базовых знаний в области алгоритмизации;
- овладение учениками основными способами представления алгоритма.

Развивающие:

- развитие у учащихся логического мышления;
- развитие способностей анализа, обобщения;
- развитие аналитических способностей.

Воспитательные:

- привлечение внимания обучающихся и их родителей к деятельности ГБНОУ СПбГЦДТТ;
- формирование творческого отношения к практической деятельности;
- формирование навыков взаимодействия в обществе;
- профориентация, профессиональная проба;
- выявление и поддержка одарённых детей;
- формирование позитивной инклюзивной среды;
- воспитание патриота и гражданина своей Родины;
- формирование интереса к ЗОЖ.

Условия организации учебно-воспитательного процесса:

Отличительной особенностью данной программы является направленность образовательного процесса на формирование способностей к алгоритмическому мышлению, крайне необходимому для дальнейшего развития учащегося в области программирования.

Программа нацелена на обучение основам алгоритмического мышления и способов разработки алгоритмов. Процесс обучения строится с учётом знаний, умений и навыков, приобретаемых школьниками на уроках в соответствии с обязательным образовательным минимумом.

В данной программе используется технология интерактивного обучения на основе использования новейших технических средств.

Оборудованы 9 рабочих мест, оснащенных персональными компьютерами с программным обеспечением, необходимым для реализации данной программы.

Контроль знаний проводится по результатам итогового контроля с помощью анализа данных.

Вводный контроль. Проводится в форме устного собеседования, направленного на выявление способностей к алгоритмическому и логическому мышлению.

Текущий и итоговый контроль состоит из теоретической и практической части.

Теоретическая:

- знание определений, связанных с алгоритмизацией,
- контроль понимания назначений и типов алгоритмов.

Практическая:

- решение задачи в среде «Scratch»,
- написание простого алгоритма с помощью псевдокода.

Требования к полученным знаниям и умениям учащихся 1 года обучения:

Обучающиеся должны овладеть следующими ключевыми компетенциями:

Личностные

- воспитание личностных качеств: трудолюбия, порядочности, ответственности, аккуратности, предприимчивости, патриотизма, а также культуры поведения и бесконфликтного общения;
- умение работать в команде;
- умение отстаивать свою позицию;
- умение нести личную ответственности за порученное дело.

Предметные

- изучение основных элементарных алгоритмов на понятийном уровне;
- умение конкретизировать и абстрагировать алгоритм;
- работа в среде программирования «Scratch» и «Python»;
- использование ПК для создания программных продуктов;
- представление алгоритм в виде псевдокода;
- анализ созданных алгоритмов.

Метапредметные

- умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий;
- умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;
- развитие логического мышления;
- развитие аналитических способностей.

СОДЕРЖАНИЕ ПРОГРАММЫ ПЕРВОГО ГОДА ОБУЧЕНИЯ

1. Введение

Теория: вводный инструктаж. Определение понятия программист, определение сферы деятельности программиста. Алгоритм, определение и основные свойства.

Практика: знакомство с учениками, вводный контроль, анализируются различные алгоритмы из повседневной жизни, желательны, предложенные самими учениками.

2. Знакомство с средой разработки Scratch и основными средствами программирования

Теория: информация по интерфейсу программной системы изучения программирования, введение основных понятий, связанных с алгоритмами: цикл, функция, условие. Виды алгоритмов.

Практика: изучение интерфейса, создание простейших алгоритмов в программной системе.

3. События, циклы и переменные

Теория: Взаимодействие между спрайтами. Циклы. Переменные, списки, использование переменных.

Практика: решение задач с использованием изучаемого материала.

4. Функции

Теория: понятие функции и переиспользования кода.

Практика: реализация серии программ с целью изучения и повторения теории.

5. Дополнительные возможности Scratch

Теория: Изучение различных дополнительных возможностей среды программирования Scratch – перо, таймер, захват видео.

Практика: реализация серии задач с использованием дополнительных возможностей.

6. Проектная деятельность

Теория: вводная информация по возможным знаниям на проект, разработка идеи проекта, возможен выбор из предложенных педагогом тем. Возможность сопровождения ПП, этапы разработки, планирование. Основы презентации программного продукта.

Практика: планирование архитектуры проекта, написание части алгоритмов. Написание основной части кода проекта. Завершение разработки, анализ созданного ПП, подготовка доклада к защите. Презентация ПП перед другими учащимися объединения.

7. Хакатон (зима)

Теория: основы проектной деятельности, экстремальное программирование.

Практика: выполнение проекта в командах.

8. Введение в Python

Теория: основы языка Python (ввод-вывод, арифметика).

Практика: решение задач на ввод-вывод и вычисления.

9. Условия, строки и циклы

Теория: Бинарные и множественные условия, логические операции, циклы for и while.

Практика: решение задач на заданную теорию.

10. Функции и структуры данных

Теория: Функции, параметры функции, рекурсия, словари, множества.

Практика: решение задач.

11. Итоговый проект

Теория: теория создания-чат ботов.

Практика: создание чат-бота на свободную тему.

12. Хакатон (лето)

Теория: углубление проектной деятельности, методологии программирования.

Практика: выполнение проекта в командах.

Календарно - тематическое планирование 1 год

№	Кол-во часов	Даты		Раздел	Тема занятия	Соответствующая конкретная тема занятия	Методическое обеспечение занятия
		по плану	по факту				
1.	2			Введение	Вводное занятие	Инструктаж по ТБ. Установочная лекция.	Инструкции по требованиям безопасности (Приложение). Текущая образовательная программа.
2.	2				Знакомство со средой Scratch	Знакомство со средой программирования "Scratch".	Программа «Scratch». Руководство пользователя.
3.	2			Основные средства программирования в Scratch	Знакомство со средой Scratch	Создание и анализ алгоритмов из жизни.	ru.wikipedia.org/wiki/Алгоритм
4.	2				Программирование движения	Игра «Программирование танца». Выполнение задания «10 блоков»	Описание игры «Программирование танца». Текст задания «10 блоков».
5.	2				Условия и сенсоры	Вводный контроль. Координатная плоскость. Команды движения.	Креативное программирование. Карен Бреннан, Кристиан Болкх, Мишель Чунг, Гарвардская Высшая школа образования [2] – с. 78. Презентация по созданию игры «Лабиринт»
6.	2				Сенсоры	Условия. Сенсоры касания.	Креативное программирование. Карен Бреннан, Кристиан Болкх, Мишель Чунг, Гарвардская Высшая школа образования [2] – с. 78. Презентация по созданию игры «Убеги от дракона»

7.	2			Сенсоры	Проверка нажатия клавиатуры и мыши. Подготовка к выставке «Наш центр вчера – сегодня - завтра»	Креативное программирование. Карен Бреннан, Кристиан Болкх, Мишель Чунг, Гарвардская Высшая школа образования [2] – с. 78. Презентация по созданию игры «Вертолёт»
8.	2			События	Управление спрайтами с клавиатуры. Подготовка к выставке «Наш центр вчера – сегодня - завтра»	Программирование для детей, Переменные – с 51 Презентация по созданию игры «Голодная рыба»
9.	2			Циклы	Простые циклы – повторить, всегда. Задание «Ганцевальная вечеринка»	Презентация, инструкция по разработке программы «Ганцевальная вечеринка».
10.	2			Циклы	Циклы «ждать до», «повторять до». Урок цифры «Искусственный интеллект в образовании»	Программирование для детей, циклы, https://урокцифры.рф
11.	2			Циклы	Костюмы спрайтов. Проект «О себе».	Презентация по созданию проекта «О себе».
12.	2			Переменные	Понятие переменных.	Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти
13.	2			Переменные	Блоки «спросить и ждать» и ответ. Чат-бот.	Презентация по созданию простого чат-бота.
14.	2			Переменные	Арифметические операции с переменными.	Презентация, инструкция по разработке программы «Арифметический тренажёр»

15.	2				Сообщения	Списки. Дебаты и разработка викторины на тему «ЗОЖ»	Программирование для детей, Кэрол Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти
16.	2				Сообщения	Сообщения. Взаимодействие между спрайтами.	Креативное программирование Карен Бреннан, Кристиан Болкх, Мишель Чунг, Гарвардская Высшая школа образования – гл. 6 – Хакатон (с. 111-112, 116, 118-119)
17.	2				Повторение изученного	Решение задач на взаимодействие	Презентация по разработке открытки в Scratch
18.	2				Ввод данных	Конструктор предложений.	Программирование для детей, Кэрол Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти
19.	2				Блоки	Понятие блоков и функций	Программирование для детей, Кэрол Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти - перо
20.	2				Блоки	Оптимизация викторины с помощью блоков.	https://www.youtube.com/watch?v=lbu4RBk3zoM
21.	2				Блоки	Создание интерфейса Интернет-магазина.	Презентацию по блокам и созданию интернет-магазина.
22.	2			Дополнительные возможности Scratch	Перо	Перо и циклы. Урок цифры «Разработка игр»	Программирование для детей, Кэрол Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти Решение и ветвление – с 68

23.	2			Дополнительные возможности Scratch	Перо	Таймер. События по времени.	Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти	
24.	2				Видео	Захват видео. Добавление эффектов. Проверка движения.	Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти Видео – с 74	
25.	2					Музыка	Работа с музыкой.	Крейг Стили, Клэр Куигли, Дэниел Маккаферти - Креативное программирование
26.	2					Синтезатор речи	Дискуссия на тему «Инклюзия». Создание синтезатора речи.	Презентация по созданию синтезатора речи.
27.	2			Проектная деятельность	Хакатон	Проекты в Scratch. Ролевая игра «Хакатон». Командообразование.	Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, – гл. 6 – Хакатон (с. 111-112, 116, 118-119) http://dou.ua/lenta/articles/scrum-for-developers/ Д. Ван Тассел Стил, разработка, эффективность, отладка и испытание программ = Programstyle, design, efficiency, debugging, andtesting. — 2-е изд. — М.: Мир, 1985. — 332 с. Презентации на каждый этап.	

28.	2				Хакатон	Генерация идеи собственного проекта.	<p>Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, – гл. 6 – Хакатон (с. 111-112, 116, 118-119) http://dou.ua/lenta/articles/scrum-for-developers/ Д. Ван Тассел Стиль, разработка, эффективность, отладка и испытание программ = Programstyle, design, efficiency, debugging, andtesting. — 2-е изд. — М.: Мир, 1985. — 332 с.</p> <p>Презентации на каждый этап. Программа «Pucharm» Программа¹ "HelloWorld" на языке Python.</p>
29.	2				Хакатон	Реализация базовых функций проекта.	<p>Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, – гл. 6 – Хакатон (с. 111-112, 116, 118-119) http://dou.ua/lenta/articles/scrum-for-developers/ Д. Ван Тассел Стиль, разработка, эффективность, отладка и испытание программ = Programstyle, design, efficiency, debugging, andtesting. — 2-е изд. — М.: Мир, 1985. — 332 с.</p>

¹ Здесь и далее программа собственного авторства. Первое использование программы — теоретический разбор. Второе использование программы — в качестве образца для написания аналогичной.

							Презентации на каждый этап. Программа «Pycharm» Программа ² "HelloWorld" на языке Python. https://stepik.org/lesson/262226/step/1?unit=243108,
30.	2				Хакатон	Продолжение реализации базовых функций проекта.	Программирование для детей, Кэрл Вордерман, Джон Вудкок, Шон Макманус, – гл. 6 – Хакатон (с. 111-112, 116, 118-119) http://dou.ua/lenta/articles/scrum-for-developers/
31.	2				Разработка архитектуры проекта	Реализация дополнительных функций проекта.	Д. Ван Тассел Стил, разработка, эффективность, отладка и испытание программ = Programstyle, design, efficiency, debugging, andtesting. — 2-е изд. — М.: Мир, 1985. — 332 с.
32.	2				Разработка проекта	Окончание реализации проекта. Промежуточный контроль.	Презентации на каждый этап. Программа «Pycharm» Программа ³ "HelloWorld" на языке Python. https://stepik.org/lesson/262226/step/1?unit=243108,
33.	2				Защиты проектов	Тестирование проекта.	
34.	2				Подготовка к защите проекта	Окончание работы над проектом. Подготовка презентации.	
35.	2				Рефлексия	Презентации и обсуждение проектов Интеллектуальная игра «Что? Где? Код!»	
36.	2			Основы Python.	Знакомство с форматом	Введение в Python. Изучение базового синтаксиса ввода-вывода. Урок цифры «Безопасность»	

37.	2				Базовый синтаксис	Понятие переменных. Ввод-вывод в переменные.	https://stepik.org/lesson/262226/step/1?unit=243108,
38.	2				Арифметика	Параметры ввода-вывода.	
39.	2				Арифметика	Сложение, вычитание, умножение	https://stepik.org/lesson/411611/step/1?unit=401045
40.	2				Арифметика	Преобразование типов данных.	https://stepik.org/lesson/411611/step/1?unit=401045
41.	2				Арифметика	Деление. Порядок выполнения операций.	https://stepik.org/lesson/411612/step/1?unit=401049
42.	2				Арифметика	Целочисленная арифметика. Нахождение остатка от деления.	https://stepik.org/lesson/416347/step/1?unit=405861
43.	2				Арифметика	Целочисленная арифметика. Обработка цифр числа.	https://stepik.org/lesson/417537/step/1?unit=407043
44.	2			Условия, строки и циклы.	Условия	Закрепление. Решение задач на арифметику.	https://stepik.org/lesson/417537/step/1?unit=407043
45.	2				Условия	Логические операции. Операторы сравнения.	https://stepik.org/lesson/262226/step/1?unit=243108: Программа «Следующее и предыдущее» на языке Python.
46.	2				Условия.	Условия. Выбор из двух ветвей.	https://stepik.org/lesson/262226/step/1?unit=243108,
47.	2				Условия	Условия. Выбор из нескольких ветвей.	https://stepik.org/lesson/262226/step/1?unit=243108, Примеры
48.	2				Условия	Вложенные условия. Каскадные условия.	https://stepik.org/lesson/262226/step/1?unit=243108, Программа «Парты»
49.	2				Строки	Закрепление. Решение задач.	https://stepik.org/lesson/262226/step/1?unit=243108, Программа «Делёж яблок» на

							языке Python: https://stepik.org/lesson/262228/step/7?unit=243110
50.	2				Строки	Строки. Операции со строками.	https://stepik.org/lesson/411612/step/1?unit=401049
51.	2				Списки	Строки. Методы find, replace, count.	https://stepik.org/lesson/411612/step/1?unit=401049
52.	2				Списки	Перебор. Цикл for.	https://stepik.org/lesson/424204/step/1?unit=413920
53.	2				Списки	Цикл while.	https://stepik.org/lesson/467730/step/1?unit=458549
54.	2				Циклы	Решение задач на циклы while и for.	https://stepik.org/lesson/424204/step/1?unit=413920
55.	2				Циклы	Списки. Массивы.	https://stepik.org/lesson/262227/step/1?unit=243109
56.	2			Функции и структуры данных	Методы массивов	Операции со списками.	https://stepik.org/lesson/262227/step/3?unit=243109
57.	2				Вложенные списки	Срезы в строках и списках.	https://stepik.org/lesson/262271/step/2?unit=243154
58.	2				Вложенные списки	Вложенные списки.	https://stepik.org/lesson/262227/step/5?unit=243109
59.	2				Повторение	Ввод вложенных списков.	Программа "Что надеть?" на языке Python: https://stepik.org/lesson/262271/step/2?unit=243154

60.	2				Функции	Повторение ввода-вывода, условий, циклов.	https://stepik.org/lesson/440666/step/1?unit=430863
61.	2				Функции	Понятие функции и процедуры.	https://stepik.org/lesson/440666/step/1?unit=430863
62.	2				Функции	Параметры функции.	https://stepik.org/lesson/440666/step/1?unit=430863
63.	2				Области видимости	Возвращаемое значение функции. Решение задач.	https://stepik.org/lesson/424204/step/1?unit=413920
64.	2				Рекурсия	Локальные и глобальные переменные.	https://stepik.org/lesson/424204/step/1?unit=413920
65.	2				Закрепление	Понятие рекурсии. Решение задач.	https://stepik.org/lesson/424204/step/1?unit=413920
66.	2				Множества	Закрепление. Решение задач на функции.	https://stepik.org/lesson/424204/step/1?unit=413920
67.	2				Словари	Множества. Решение задач.	https://stepik.org/lesson/424204/step/1?unit=413920
68.	2				Повторение	Словари. Решение задач.	https://stepik.org/lesson/424204/step/1?unit=413920
69.	2				Чат-бот	Повторение изученного. Решение задач.	https://stepik.org/lesson/424204/step/1?unit=413920
70.	2				Чат-бот	Итоговый контроль. Понятие чат-бота. Начало работы над чат-ботом.	https://stepik.org/lesson/424204/step/1?unit=413920
71.	2				Чат-бот	Задание команд чат-бота.	Презентация по теме «Разработка чат-бота на Python»
72.	2				Чат-бот	Подключение API к чат-боту.	Презентация по теме «Разработка чат-бота на Python»
73.	2			Хакатон	Чат-бот	Окончание работы над чат-ботом. Подготовка презентации.	Презентация по теме «Разработка чат-бота на Python»

74.	2			Практика программирования	Презентация чат-бота. Рефлексия. Открытое занятие.	Конспект защиты. Дональд Кнут Искусство программирования, том 1. Основные алгоритмы = The Art of Computer Programming, vol.1. Fundamental Algorithms. — 3-е изд. — М.: «Вильямс», 2006. — С. 720. — ISBN 0-201-89683-4
75.	2			Практика программирования	Реализация базовых функций проекта.	
76.	2			Практика программирования	Продолжение работы над MVP проекта	
77.	2			MVP	Окончание работы над MVP проекта	
78.	2			Практика программирования	Презентация MVP. Продолжение работы.	
79.	2			Практика программирования	Расширение функционала проекта	
80.	2			Практика программирования	Подготовка презентации проекта.	
Итого: 160 часов						

Утверждаю
Директор СПбГЦДТТ
_____ А.Н. Думанский

РАБОЧАЯ ПРОГРАММА

к дополнительной общеразвивающей программе

«КРЕАТИВНОЕ ПРОГРАММИРОВАНИЕ (Scratch+Python)»

20__ - 20__ учебный год

Год обучения 2

Группа № _____

Румянцева Мария Юрьевна,
педагог дополнительного образования
СПбГЦДТТ

Рабочая программа 2-го года обучения составлена на основе дополнительной общеразвивающей программы «Креативное программирование (scratch+python)» технической направленности.

Цель второго года обучения: Сформировать у учащихся системы компетенций, связанных со знаниями в области алгоритмизации, общих закономерностей развития IT-области для дальнейшего их применения при освоении различных областей программирования и выполнения работ, связанных с программированием и алгоритмизацией.

Задачи 2 года обучения:

Обучающие:

- изучение способа представления алгоритма «Псевдокод»;
- повторение элементарных алгоритмов и их представления в псевдокоде;
- написание программ на базе языка программирования Python;
- изучение теоретических основ программирования;
- приобретение учениками базовых знаний в области алгоритмизации;
- овладение учениками основными способами представления алгоритма.

Развивающие

- развитие у учащихся логического мышления;
- развитие способностей анализа, обобщения;
- развитие аналитических способностей.

Воспитательные

- привлечение внимания обучающихся и их родителей к деятельности ГБНОУ СПбГЦДТТ;
- формирование творческого отношения к практической деятельности;
- формирование навыков взаимодействия в обществе;
- профориентация, профессиональная проба;
- выявление и поддержка одарённых детей;
- формирование позитивной инклюзивной среды;
- воспитание патриота и гражданина своей Родины;
- формирование интереса к ЗОЖ.

Условия организации учебно-воспитательного процесса:

Отличительной особенностью данной программы является направленность образовательного процесса на формирование способностей к алгоритмическому мышлению, крайне необходимому для дальнейшего развития учащегося в области программирования.

Программа нацелена на обучение основам алгоритмического мышления и способов разработки алгоритмов. Процесс обучения строится с учётом знаний, умений и навыков, приобретаемых школьниками на уроках в соответствии с обязательным образовательным минимумом.

В данной программе используется технология интерактивного обучения на основе использования новейших технических средств.

Оборудованные 9 рабочих мест, оснащенных персональными компьютерами с программным обеспечением, необходимым для реализации данной программы.

Контроль знаний проводится по результатам итогового контроля и анализа данных.

Вводный контроль полностью повторяет итоговый контроль первого года обучения.

Итоговый контроль состоит из теоретической и практической части.

Теоретическая часть направлена на контроль теоретических знаний по всему курсу обучения.

Практическая:

- решение задачи с помощью Блок-схем;
- написание программы на языке Python.

ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ

Обучающиеся должны овладеть следующими ключевыми компетенциями:

Личностные

- воспитание личностных качеств: трудолюбия, порядочности, ответственности, аккуратности, предприимчивости, патриотизма, а также культуры поведения и бесконфликтного общения;
- умение работать в команде;
- умение отстаивать свою позицию;
- умение нести личную ответственности за порученное дело.

Предметные

- изучение основных элементарных алгоритмов на понятийном уровне;
- умение конкретизировать и абстрагировать алгоритм;
- работа в среде программирования «Scratch»;
- использование ПК для создания программных продуктов;
- представление алгоритм в виде псевдокода;
- анализ созданных алгоритмов;
- основные термины алгоритмизации и программирования;
- знать и понимать цели каждого из этапов создания программного продукта;
- основные принципы написания «Красивого» программного кода.

Метапредметные

- умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий;
- умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;
- развитие логического мышления;
- развитие аналитических способностей.

СОДЕРЖАНИЕ ПРОГРАММЫ ВТОРОГО ГОДА ОБУЧЕНИЯ

1. Вводное занятие.

Теория: вводный инструктаж.

Практика: знакомство с учениками, вводный контроль.

2. Ввод-вывод, арифметика и элементарные вычисления.

Теория: знакомство с интерфейсом среды программирования. Изучение основ синтаксиса языка программирования. Основные приёмы человеко-машинного взаимодействия. Понятие переменных.

Практика: знакомство с интерфейсом, знакомство с документацией. Анализ простейшего примера программы на языке программирования под руководством педагога, запуск этой программы учениками. Решение задач.

3. Базовые элементы синтаксиса

Теория: определение, основные принципы, основные конструкции: последовательное исполнение, ветвление, цикл, пошаговость разработки. Подробный анализ основных свойств и методов подхода.

Практика: анализ нескольких алгоритмов с точки зрения структурного программирования. Решение задач, заостряя внимание на структурном подходе.

4. Практика программирования

Теория: основные правила написания красивого и хорошо читаемого кода, основы написания чат-ботов.

Практика: реализация и публикация чат-бота

5. Продвинутое элементы синтаксиса

Теория: алгоритмическое введение в задачу, повторение математических основ, используемых в задаче.

Практика: написание программ в соответствии с заданием.

6. ООП и библиотека Pygame

Теория: вводная информация по объектно-ориентированному программированию, основная информация по библиотеке Pygame.

Практика: Написание программ с использованием библиотеки Pygame. Решение задач. Создание небольшой простой игры.

7. Разработка собственного проекта

Теория: вводная информация по возможным знаниям на проект, разработка идеи проекта, возможен выбор из предложенных педагогом тем. Возможность сопровождения ПП, этапы разработки, планирование. Основы презентации программного продукта.

Практика: планирование архитектуры проекта, написание части алгоритмов. Написание основной части кода проекта. Завершение разработки, анализ созданного ПП, подготовка доклада к защите. Презентация ПП перед другими учащимися объединения

Календарно – тематическое планирование 2 год

№ занятия	Кол-во часов	Даты		Раздел	Тема занятия	Соответствующая конкретная тема занятия	Методическое обеспечение занятия
		по плану	по факту				
1.	2			Вводное занятие.	Вводное занятие	Инструктаж по ТБ. Установочная лекция.	Инструкции по требованиям безопасности (Приложение). Текущая образовательная программа, задания вводного контроля.
2.	2				Вводное занятие	Постановка целей на курс. Программа «Обо мне».	
3.	2			Повторение	Ввод-вывод	Повторение и усложнение базового синтаксиса ввода-вывода.	Программа «Pycharm» Программа ⁴ "HelloWorld" на языке Python.
4.	2				Переменные	Повторение и усложнение понятия переменных.	https://stepik.org/lesson/262226/step/1?unit=243108 , Программа "Hello, Harry!" на языке Python: https://stepik.org/lesson/262228/step/5?unit=243110
5.	2				Повторение арифметики.	Сложение, вычитание, умножение. Урок цифры «Искусственный интеллект в образовании»	

⁴ Здесь и далее программа собственного авторства. Первое использование программы — теоретический разбор. Второе использование программы — в качестве образца для написания аналогичной.

6.	2				Повторение арифметики.	Деление. Остаток от деления.	https://stepik.org/lesson/262226/step/1?unit=243108: Программа «Следующее и предыдущее» на языке Python.
7.	2				Повторение условий.	Комплексные условия.	https://stepik.org/lesson/262226/step/1?unit=243108,
8.	2				Повторение циклов	Циклы и вложенные циклы. Дебаты «За и против ЗОЖ». Создание викторины.	https://stepik.org/lesson/262226/step/1?unit=243108, Примеры
9.	2				Повторение массивов	Массивы. Срезы. Поиск.	https://stepik.org/lesson/262226/step/1?unit=243108, Программа «Парты»
10.	2				Повторение строк	Строки. Дискуссия на тему «Инклюзия». Тематическая практика.	https://stepik.org/lesson/262226/step/1?unit=243108, Программа «Делёж яблок» на языке Python: https://stepik.org/lesson/262228/step/7?unit=243110
11.	2				Арифметика	Функции. Решение задач.	https://stepik.org/lesson/411612/step/1?unit=401049
12.	2				Арифметика	Возвращаемое значение функции.	https://stepik.org/lesson/411612/step/1?unit=401049
13.	2				Арифметика	Локальные и глобальные переменные.	https://stepik.org/lesson/411612/step/7?unit=401049
14.	2				Арифметика	Введение в классы.	https://stepik.org/lesson/416347/step/1?unit=405861

15.	2				Арифметика	Конструкторы классов.	https://stepik.org/lesson/416347/step/1?unit=405861
16.	2			Базовые элементы синтаксиса	Условия	Наследование.	https://stepik.org/lesson/262227/step/1?unit=243109
17.	2				Условия	Практика использования классов.	https://stepik.org/lesson/262227/step/3?unit=243109 Программа «Минимум из двух чисел» на языке Python
18.	2				Логические операции	Словари. Создание словаря.	https://stepik.org/lesson/262227/step/2?unit=243109
19.	2				Вложенные условия	Словари. Перебор элементов словаря.	https://stepik.org/lesson/262227/step/5?unit=243109
20.	2				Каскадные условия	Вложенные списки. Решение задач.	https://урокцифры.рф/lesson/bolshie-dannye/
21.	2				Типы данных	Обработка двумерных массивов. Решение задач	Пример программы с использованием цикла while.
22.	2				Срезы	Работа с API.	https://stepik.org/lesson/440666/step/1?unit=430863
23.	2				Методы массивов	Парсинг сайтов.	https://stepik.org/lesson/440666/step/1?unit=430863
24.	2				Методы массивов	Парсинг сайтов.	https://stepik.org/lesson/440666/step/1?unit=430863
25.	2				Разработка РПГ	Продумывание игрового мира и персонажей.	Презентация
26.	2				Разработка РПГ	Создание каркаса текстовой игры.	Презентация
27.	2				Разработка РПГ	Программирование основного сценария.	Презентация
28.	2				Разработка РПГ	Программирование жизни.	https://stepik.org/lesson/424204/step/1?unit=413920
29.	2			Разработка РПГ	Программирование	https://stepik.org/lesson/424204/step/	

					кошелька.	1?unit=413920
30.	2				Разработка РПГ	Сборка и тестирование. https://stepik.org/lesson/424204/step/1?unit=413920
31.	2				Разработка РПГ	Презентация. https://stepik.org/lesson/424204/step/1?unit=413920
32.	2				Чат-бот	Понятие чат-бота. Начало работы над чат-ботом. https://stepik.org/lesson/424204/step/1?unit=413920
33.	2				Чат-бот	Подключение API к чат-боту. https://stepik.org/lesson/424204/step/1?unit=413920
34.	2				Чат-бот	Окончание работы над чат-ботом. Подготовка презентации. https://stepik.org/lesson/424204/step/1?unit=413920
35.	2				Чат-бот	Презентация чат-бота. Игра «Что? Где? Код!»
36.	2			Хакатон	Графический интерфейс	Инструктаж по ТБ. Презентация по теме «Разработка чат-бота на Python»
37.	2				Чат-бот	Введение в графический интерфейс. Презентация по теме «Разработка чат-бота на Python»
38.	2				Функции	Виджет кнопок, текста. Презентация по теме «Функции»
39.	2			Продвинутые элементы синтаксиса	Функции	Поле ввода. Презентация по теме «Функции»
40.	2				Функции	Метод раск. Задача «Переставить два слова»
41.	2				Функции	Поле Text.
42.	2				Рекурсия	Переключатели Задача «Первое и последнее вхождения»

43.	2			Функции	Список выбора	Инструкции по требованиям безопасности (Приложение)
44.	2			Двумерные массивы.	Связывание переменных и виджетов.	Презентация «Списки»
45.	2			Вложенные списки.	События.	Презентация «Списки»
46.	2			Обработка списков.	Поле рисования. Начало проекта «Рисовалка»	Презентация «Списки»
47.	2			Множества	Анимация рисования. Продолжение проекта.	Презентация «Списки»
48.	2			Множества	Виджет меню.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 5.1, 5.2, 5.7.
49.	2			Словари	Окончание проекта «рисовалка».	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 5.13, 5.14, 5.23.
50.	2			Словари	Библиотека rугame. Каркас игры на rугame	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.1, 11.7.
51.	2			Решение задач	Игровой цикл.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил.

							Задание 11.14, 11.19	
52.	2				Понятие ООП	Геометрические примитивы в ругаме.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19	
53.	2			ООП и библиотека Rугame.	Конструкторы	Обработка событий клавиатуры.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19	
54.	2				Классы	Обработка событий мыши.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19	
55.	2					Каркас игры	Создание поверхностей. Класс surface.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19
56.	2					Примитивы	Работа с классом Rect.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19
57.	2					События	Работа с текстом. Решение задач.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19

58.	2				Классы rугame	Работа с изображением. Решение задач.	Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил. Задание 11.14, 11.19
59.	2				Квадраты	Спрайты.	
60.	2				Текст	Группы спрайтов.	
61.	2				Изображения	Звуки игры.	
62.	2				Спрайты	Сборка игры.	
63.	2				Проект	Основы архитектуры ПО. Создание архитектуры своего проекта.	
64.	2				Игровой мир	Разработка игрового мира.	
65.	2			Проектная деятельность	Карта	Разработка карты.	http://www.kv.by/index2008354201.htm http://dou.ua/lenta/articles/scrum-for-developers/
66.	2				Персонажи	Разработка персонажей.	http://ru.wikipedia.org/wiki/Архитектура_программного_обеспечения Дональд Кнут Искусство программирования, том 1.
67.	2				Персонажи	Программирование персонажей.	Д. Ван Тассел Стил, разработка, эффективность, отладка и испытание программ = Programstyle, design, efficiency, debugging, andtesting. — 2-е изд. — М.: Мир, 1985. — 332 с.
68.	2				Персонажи	Реализация управления	Дональд Кнут Искусство

					персонажами.	программирования, том 1. Основные алгоритмы =	
69.	2				Персонажи	Реализация счёта и жизней. Задания итогового контроля.	
70.	2				Презентация	Окончание работы над проектом. Подготовка презентации.	
71.	2				Презентация	Защита проектов учащимися. Обсуждение проектов.	
72.	2			Хакатон	Выполнение проекта в командах.	Дональд Кнут Искусство программирования, том 1. Основные алгоритмы = The Art of Computer Programming, vol.1. Fundamental Algorithms. — 3-е изд. — М.: «Вильямс», 2006. — С. 720. — ISBN 0-201-89683-4	
73.	2				Практика программирования		Проектирование архитектуры и дизайна проекта.
74.	2				Практика программирования		Реализация базовых функций проекта.
75.	2				Практика программирования		Продолжение работы над MVP проекта
76.	2				MVP		Окончание работы над MVP проекта
77.	2				Практика программирования		Презентация MVP. Продолжение работы.
78.	2				Практика программирования		Усложнение кода проекта.
79.	2				Практика программирования		Добавление дополнительного функционала.
80.	2				Практика программирования		Расширение функционала проекта
					Итого: 166 часов		

Педагогические методики и технологии

Процесс достижения поставленных целей и задач программы осуществляется в сотрудничестве учащихся и педагога. При этом реализуются различные методы осуществления целостного педагогического процесса. На различных его этапах ведущими выступают отдельные, приведенные ниже методы:

Методы организации учебно-познавательной деятельности:

- Словесные - беседа, рассказ, монолог, диалог,
- Наглядные - демонстрация готовых сайтов и т.д.,
- Практические - решение творческих заданий и др.,
- Репродуктивные - работа по шаблонам,
- Проблемно –поисковые- изготовление страниц по собственному замыслу, решение творческих задач,
- Индивидуальные - задания в зависимости от достигнутого уровня развития учащегося).

Метод проектов используется на занятиях в течение всего периода обучения. Он способствует включению подростков в проектную культуру, формированию учащимися адекватной самооценки.

Методы стимулирования и мотивации учебно-познавательной деятельности (познавательные и развивающие игры, экскурсии, коллективные обсуждения и т.д.).

Методы воспитания: беседы, метод примера, педагогическое требование, создание воспитательных ситуаций, соревнование, поощрение, наблюдение, анкетирование, анализ результатов.

Выбор метода обучения зависит от содержания занятия, уровня подготовки и опыта учащихся.

На занятиях 1 и 3 блока программы преобладают методы: беседа, рассказ, диалог, выполнение упражнений.

На занятиях 2 и 4 блока программы — метод проектов, консультирования.

Основным методом проведения занятий является практическая работа, так как именно практика позволяет наиболее полно понять основные принципы алгоритмирования и начать постигать искусство программирования. На занятиях по всем темам проводится инструктаж по технике безопасности при работе с ПК.

Решению воспитательных задач, поставленных в программе, способствуют экскурсии на выставки смежных с компьютерными технологиями тематиками Санкт-Петербурга, например «Музей высоких технологий». Большое воспитательное воздействие оказывает также участие учащихся в выставках, конкурсах различного уровня.

Дидактический материал

- Инструкции по выполнению практических заданий по каждой теме занятия в формате презентаций.
- Распечатки текстов программ.
- ГОСТ 19.701-90. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
- ГОСТ 19.003-80. Схемы алгоритмов и программ. Обозначения условные, графические.
- Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240 с.: ил.

Информационные источники

Литература для преподавателей

1. Вирт Н. Алгоритмы + структуры данных = программы. — М.: Мир, 2013;
2. Креативное программирование.
3. Программирование для детей, Кэрол Вордерман, Джон Вудкок, Шон Макманус, Крейг Стили, Клэр Куигли, Дэниел Маккаферти
4. В.Г. Рындак, В.О.Дженжер, Л.В.Денисова. Проектная деятельность школьника в среде программирования Scratch. Учебно-методическое пособие. Оренбург — 2009.

Литература, рекомендуемая для учащихся:

1. Сборник задач по программированию. — 2-е изд., перераб. И доп. — СПб.: БХВ-Петербург, 2007. — 240с.
2. Голиков Д. Scratch для юных программистов. – СПб.: БХВ-Петербург. – 2017. – 192 с
3. Python для детей

Интернет-источники

1. <https://scratch.mit.edu>
2. <http://younglinux.info/scratch>
3. <https://pythontutor.ru>
4. <http://itgenio.by>
5. <http://code.org>
6. <http://codecombat.com>
7. <http://coderussia.ru>