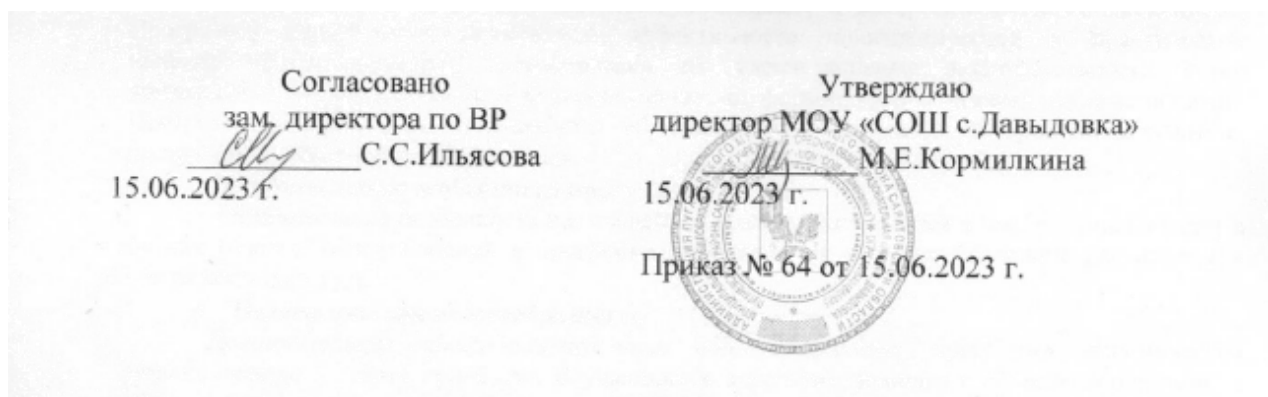


**Муниципальное общеобразовательное учреждение
«Средняя общеобразовательная школа с. Давыдовка
Пугачёвского района Саратовской области»**



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА ТЕХНИЧЕСКОЙ
НАПРАВЛЕННОСТИ
«Учимся программировать»**

Возраст учащихся:

14-18 лет

Срок реализации:

0,5 года

Автор-составитель:

Тишакова В. И.,

педагог дополнительного
образования

2023 год

1 раздел

Комплекс основных характеристик

1. Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) программа «**Учимся программировать**» составлена на основе курса Д.П. Кириенко «Основы языка программирования Python», М.: Бином, 2018 г.

Программа построена таким образом, чтобы помочь учащимся заинтересоваться программированием, найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации; при решении практических и жизненных задач. Программа строится при обучении детей так, что позволяет создавать собственные программы для решения конкретной задачи. Это является **отличительной особенностью** данной программы.

Направленность программы – техническая.

Актуальность программы

Python – это язык программирования общего назначения, распространяемый с открытыми исходными текстами. Он оптимизирован для создания качественного программного обеспечения. Язык Python используется сотнями тысяч разработчиков по всему миру в таких областях, как создание веб-сценариев, системное программирование, создание пользовательских интерфейсов, настройка программных продуктов под пользователя, численное программирование и в других.

Сегодня Python один из самых популярных языков программирования, области его применения только расширяются. Последние несколько лет он входит в ТОП-3 самых востребованных языков на рынке IT.

Также по информации порталов по трудоустройству в России в течение последних 5 лет Python-разработчики востребованы на рынке труда, специалистов в этой сфере до сих пор не хватает.

Освоив данную программу обучения, в будущем школьники приобретут престижную профессию, востребованную на сегодняшнем рынке труда. В этом и заключается **актуальность** данной программы.

Отличительные особенности данной программы

На сегодняшний день существует множество языков программирования, каждый из которых имеет свои особенности. Но хочется выделить Python, как популярную универсальную среду разработки программного кода с тридцатилетней историей. Понятный и простой язык программирования дает возможность освоить данный курс учащимся разных возрастов, не имеющим знаний в программировании. Можно создать интересные приложения, и при этом не придется сидеть неделями, изучая сложный синтаксис. Основное внимание в

курсе уделяется общим вопросам построения алгоритмов, навыкам программирования на языке Python. Такой подход к построению курса обусловлен тем, что современные компьютерные технологии с их достаточно простым пользовательским интерфейсом способствуют формированию "потребительского" отношения к ним – подрастающее поколение может стать поколением "продвинутых пользователей". Без знаний основ алгоритмизации и программирования школьнику не удастся стать хорошим программистом. Не секрет, что многие начинающие разработчики испытывают сложности именно при разработке интерфейсов и, особенно, при написании программного кода. Данная программа обучения построена таким образом, что позволяет добиться того, что юные разработчики не будут испытывать этих сложностей. В этом и заключается одна из отличительных особенностей данной программы.

Еще одной отличительной особенностью данной программы является то, что программа «**Программируем на языке Python**» в большинстве своем состоит из практических заданий, поскольку процесс усвоения нового у детей происходит лучше всего на практике. При этом каждый урок курса содержит теоретические материалы, необходимые для осмысленного выполнения практических заданий. Методические пособия для слушателей по каждому модулю оформлены в виде полноценного курса, ориентированного на изучение и выполнение конкретных задач (тем), получение конкретных навыков программирования на Python. В данной программе уделяется внимание технологическим инновациям и работе с ними.

Педагогическая целесообразность программы состоит в том, что её реализация позволяет повысить эффективность познавательного процесса обучающихся. Программа позволяет обучающемуся раскрывать в себе творческие возможности. Изучая простые команды программы, ребята учатся программировать, развивают мышление, фантазию.

Новизна программы. Какую бы сферу жизнедеятельности человека мы не взяли: медицину, проектирование зданий, машин, образование – без применения компьютерных технологий нигде в современном мире не обходится. Для каждой из этих областей разрабатываются соответствующие программы. Следовательно, сегодня является востребованной такая профессия, как программист: навыки программирования пользуются высоким спросом, должность программиста хорошо оплачивается. Даже за пределами IT-мира знание хотя бы одного языка программирования – это серьезный плюс в резюме. Курс способствует профессиональной ориентации ребенка.

Интегрирование различных школьных предметов в учебном курсе «**Программируем на языке Python**» открывает новые возможности для реализации новых образовательных концепций, овладения новыми навыками и

расширения круга интересов.

Адресат программы.

Обучающиеся 14-18 лет.

Срок реализации общеобразовательной (общеразвивающей) программы – 0,5 года.

Объем программы – 36 часов.

Формы и режим занятий. В данной программе используется индивидуальная, групповая и фронтальная формы работы.

Содержание практических занятий ориентировано не только на овладение учащимися навыками программирования, но и на подготовку их как грамотных пользователей ПК; формированию навыков участия в дистанционных конкурсах и олимпиадах, умений успешно использовать навыки сетевого взаимодействия.

Режим занятий: 1 раз в неделю по 2 часа.

Количество обучающихся в группе – 10-15 человек.

2. Цель общеобразовательной (общеразвивающей) программы – способствовать формированию 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.

3. Задачи программы:

Обучающие:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- познакомить с основными конструкциями языка программирования Python;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Развивающие:

- развивать познавательные процессы (внимание, восприятие, логическое мышление, память);
- развивать креативность;
- развивать алгоритмическое мышление;
- развивать способности к самореализации.

Воспитательные:

- формировать уважительное отношение к взрослым людям и сверстникам, культуру поведения во время занятий и совместной продуктивной деятельности;

- формировать культуру занятий, направленную на воспитание личностных и социальных качеств;
- формировать чувства коллективизма и взаимовыручки.

4. Планируемые результаты и способы определения их результативности.

Предметные результаты:

- формирование знаний и умений работы с языком программирования Python;
- формирование представления об основных изучаемых понятиях — «информация», «алгоритм», «модель» — и их свойствах;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для конкретного исполнителя; формирование знаний об алгоритмических конструкциях, логических значениях и операциях; знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- владение универсальным языком программирования высокого уровня Python, представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции;
- владение навыками и опытом разработки программ в среде программирования Python, включая тестирование и отладку программ; владение элементарными навыками формализации прикладной задачи и документирования программ;
- формирование умения работать с библиотеками программ; получение опыта использования компьютерных средств представления и анализа данных.

Метапредметные результаты:

- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности. Учащиеся научатся адекватно воспринимать содержательную оценку своей работы преподавателем и при необходимости исправлять ошибки.
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

Личностные результаты:

- учащиеся демонстрируют интерес к программированию в целом и к разработке Python-приложений в частности;
- учащиеся демонстрируют упорство в достижении поставленных целей, ответственность.

5. Учебный план

№	Раздел	Всего	Теория	Практика	Формы аттестации
1	Раздел 1. Введение в язык программирования Python.	3	1	2	Практическая работа, творческая работа, тест
2	Раздел 2. Линейные алгоритмы и их реализация на языке Python	3	1	2	Практическая работа, творческая работа, тест
3	Раздел 3. Разветвляющиеся алгоритмы	4	1	3	Практическая работа, творческая работа, тест
4	Раздел 4. Циклические алгоритмы и их реализация на Python.	8	3	5	Практическая работа, творческая работа, тест
5	Раздел 5. Этапы решения задач на Python.	1,5	0,5	1	Практическая работа, творческая работа, тест
6	Раздел 6. Вспомогательные алгоритмы и их реализация на Python.	3,5	1,5	2	Практическая работа, творческая работа, тест
7	Раздел 7. Структурные типы данных. Списки.	4	1	3	Практическая работа, творческая работа, тест
8	Раздел 8. Структурные типы данных. Символьные строки	3	1	2	Практическая работа, творческая работа, тест
9	Раздел 9. Структурные типы данных. Матрицы	3	1	2	Практическая работа, творческая работа, тест
10	Раздел 10. Структурные типы данных. Работа с файлами	1,5	0,5	1	Практическая работа, творческая работа, тест
11	Раздел 11. Итоги обучения по курсу.	1,5	0,5	1	Практическая работа, творческая работа, тест
	Итого	36	12	24	

6. Содержание программы.

Раздел 1. Введение в язык программирования Python.

Теоретическая часть: Понятие алгоритма и программы. Элементы языка. Структура программы. Операции и переменные. Типы данных. Знакомство с основными типами переменных, синтаксисом языка программирования, основными процедурами ввода исходных данных и вывода результатов. Ввод и вывод данных на языке Python. Анализ возможных синтаксических ошибок

Практическая часть: Установка и работа в среде программирования Python. Использование основных процедур ввода исходных данных и вывода результатов. Создание элементарных программ ввода-вывода данных, работа со средой, отладка программ.

Раздел 2. Линейные алгоритмы и их реализация на языке Python.

Теоретическая часть: Линейные алгоритмы целочисленных данных и их реализация на Python. Знакомство со структурой линейного алгоритма, правилами записи арифметических выражений. Различные типы данных, допустимые операции над ними и ресурсы оперативной памяти. Выполнение операции присваивания. Решение задач по теме «Обработка целочисленных данных». Анализ готовых линейных программ. Различные типы данных, допустимые операции над вещественными числами и ресурсы оперативной памяти.

Практическая часть: создание блок-схемы линейного алгоритма, разработка линейных алгоритмов. Ввод и отладка программ, реализующих линейный алгоритм обработки целых чисел. Создание блок-схем алгоритма. Ввод и отладка программ, реализующих линейный алгоритм обработки вещественных чисел. Составление алгоритма, написание программы, ввод и отладка программного кода, анализ результатов.

Раздел 3. Разветвляющиеся алгоритмы и их реализация на языке Python

Теоретическая часть: Понятие алгоритма «Выбор», графическое изображение. Типовые задачи, использующие алгоритм «Выбор», изображение алгоритма в виде блок-схемы. Составление и анализ алгоритмов. Сложные условия. Каскадные ветвления. Понятие множественного выбора, изображение на блок-схеме.

Практическая часть: Составление алгоритма «Полное ветвление», «Неполное ветвление», «Сложное ветвление». Ввод и отладка программ в среде Python.

Раздел 4. Циклические алгоритмы и их реализация на языке Python.

Теоретическая часть: Циклические алгоритмы. Понятие цикла с предусловием, графическое изображение. Типовые задачи, решаемые с помощью цикла с предусловием. Анализ составленного алгоритма. Понятие цикла с

постусловием, графическое изображение. Типовые задачи, решаемые с помощью цикла с постпроверкой. Понятие цикла с параметром, графическое изображение. Разработка алгоритмов цикла с параметром, анализ логических ошибок. Оценка эффективности разработанного алгоритма. Понятие вложенного цикла, графическое изображение.

Практическая часть: Составление блок-схем, написание программного кода, ввод и отладка программ, реализующих цикл с предусловием в среде Python. Ввод и отладка программ, реализующих цикл с постусловием в среде Python. Ввод и отладка программ, реализующих цикл с параметром в среде Python, Поиск синтаксических и логических ошибок в программах. Ввод и отладка программ, реализующих вложенные циклы в среде Python.

Раздел 5. Этапы решения задач на языке Python.

Теоретическая часть: Этапы решения задачи на компьютере. Последовательное конструирование алгоритма. Разработка алгоритмов.

Практическая часть: Написание программного кода, ввод и отладка программ в среде Python.

Раздел 6. Вспомогательные алгоритмы и их реализация на языке Python.

Теоретическая часть: Понятие вспомогательного алгоритма. Формат записи вспомогательного алгоритма в виде функции. Типовые задачи. Анализ разработанных алгоритмов. Понятие рекурсии. Типовые задачи.

Практическая часть: Ввод и отладка программ с использованием функции в среде Python. Ввод и отладка программ с использованием рекурсии в среде Python

Раздел 7. Структурные типы данных. Списки

Теоретическая часть: Понятие списка. Заполнение списка значениями. Вывод списка на экран. Обработка списков. Линейный поиск. Добавление и удаление элементов. Копирование списков. Методы поиска и сортировки в списках.

Практическая часть: Ввод и отладка программ со списками. Ввод и отладка программ по линейному поиску, добавлению и удалению элементов, копированию списков. Создание блок-схем алгоритма, ввод и отладка программ по поиску и сортировке в списках

Раздел 8. Структурные типы данных. Символьные строки.

Теоретическая часть: Понятие символьных строк. Основные операции обработки. Характеристика и классификация типовых задач по обработке строк. Составление и анализ алгоритмов

Практическая часть: Решение задач по обработке строк, ввод и отладка программ.

Раздел 9. Структурные типы данных. Матрицы.

Теоретическая часть: Понятие матрицы. Методы ввода элементов. Анализ типовых задач обработки элементов матрицы. Обработка исключений.

Практическая часть: Разработка алгоритмов. Ввод и отладка программ в Python. Анализ синтаксических и логических ошибок.

Раздел 10. Структурные типы данных. Работа с файлами.

Теоретическая часть: Типы файлов, этапы обработки, режимы открытия. Программирование файлового ввода/вывода. Текстовые и двоичные файлы. Этапы обработки файла. Программирование обработки файлов. Методы считывания данных из файла. Типовые задачи по изменению файлов.

Практическая часть: Составление алгоритмов, написание программного кода, ввод и отладка программ в среде Python.

Раздел 11. Итоги обучения по курсу.

Теоретическая часть: Анализ результатов работы. Обзор пройденного материала.

Практическая часть: составление алгоритма, написание программы, ввод и отладка программного кода. Анализ выполненных проектов.

Способы определения результативности реализации программы.

Знания, умения, навыки, полученные на занятиях, необходимо подвергать педагогическому контролю, с целью выявления качества усвоенных детьми знаний в рамках программы обучения.

Формами педагогического контроля могут быть: итоговые занятия один раз в полугодие, контрольные задания, тематические выставки, устный опрос, тестирование, которые способствуют поддержанию интереса к работе, направляют учащихся к достижению более высоких вершин творчества.

7. Формы организации контроля

- **Входящий контроль** осуществляется при комплектовании группы в начале учебного года. Цель – определить исходный уровень знаний обучающихся, определить формы и методы работы с ними. 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 разноуровневых заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу обучающихся, позволяют более полно проявлять

полученные знания, умения, навыки.

- **Промежуточный контроль** осуществляется конце декабря-начале января. Форма контроля: тест, решение практических задач средствами языка программирования. Итоговый контроль осуществляется в конце мая. Форма контроля: защита творческого проекта.

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

При безоценочной системе курса итоговый контроль реализован в форме итогового задания или теста.

Методы определения результата:

- педагогическое наблюдение;
- оценка продуктов творческой деятельности обучающихся;
- беседы, тесты, опросы.

Формы определения результата:

- выполнение зачетных заданий по пройденным темам;
- участие в проектной деятельности.

Критерии оценивания обучающихся по курсу:

- на курсе дополнительного образования «программируем на языке python» действует безоценочная система;

- содержание и объем материала, подлежащего проверке, определяется программой;

- при проверке усвоения материала необходимо выявлять полноту, прочность усвоения обучающимися теории и умение применять ее на практике в знакомых и незнакомых ситуациях;

- основными формами проверки являются завершённые практические работы, тестирование, самостоятельная работа, устный опрос.

Формы подведения итогов:

Результативность обучения обеспечивается применением различных форм, методов и приемов, которые тесно связаны между собой и дополняют друг друга. В ходе практической деятельности педагог тактично контролирует, советует, направляет обучающихся. Большая часть занятий отводится практической работе, по окончании которой" проходит обсуждение и анализ.

2 раздел

Комплекс организационно – педагогических условий

1. Методическое обеспечение

Учебное пособие «Программирование. Python, C++» для 8–11 классов. Пособие выходит в издательстве «БИНОМ. Лаборатория знаний» в 2018 году. Предназначено для изучения программирования.

Пособие включает четыре части (четыре уровня), которые могут быть использованы для проведения занятий в 8-11 классах.

Методика преподавания курса предусматривает:

- проведение по каждой новой теме лекции
- выполнение обучающимися самостоятельного практического задания на каждом занятии;
- уровень задания определяется педагогом для каждого ребенка индивидуально, с учетом его возможностей, интересов и склонностей; при самостоятельном изучении курса, обучающийся сам выбирает уровень практической работы;
- для промежуточного и итогового контроля используются тестовые задания; Прохождение курса сопровождается созданием обучающимися проектов по предлагаемым темам. Подобная организация обучения способствует решению задачи развития познавательной активности и творческих способностей обучающихся.

2. Условия реализации программы

Программа реализуется в очной и очно-заочной форме.

• **Материально-техническое обеспечение программы:**

Для успешной реализации программы создаются условия необходимые для реализации программы в течение всего периода, а именно, кабинет, соответствующий требованиям реализации задач «Точки роста», оборудование «Точки роста», программное обеспечение.

- **Учебно – дидактические материалы** (для педагога, обучающихся, родителей): специализированная литература, подборка интернет – ресурсов, наборы технической документации к применяемому оборудованию, образцы программ, выполненные обучающимися и педагогом, учебно – методические пособия для педагога и обучающихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное оборудование.

Список литературы и ЭОР для педагога:

- Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2020. – 284 с.: ил.
- Программирование на языке Python: учебный курс.: Пер. с англ. – СПб.: ООО "Альфа-книга": 2020. – 736 с.
- Тарасов Д. Факультативный курс «Основы алгоритмизации и программирования на языке Python» 10-11 классы. Видеоуроки для учителей, родителей и учеников.

Список литературы и ЭОР для обучающегося:

- Босова Л.Л. Информатика. 8-9 классы. Начала программирования на языке Python. Дополнительные главы к учебникам. – БИНОМ. Лаборатория знаний, 2020. – 96с.
- Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2020. – 284 с.: ил.
- Поляков К. Ю, Еремин В.А. Информатика. 10 класс. Учебник. Углубленный уровень. В 2 частях. ФГОС. – БИНОМ. Лаборатория знаний, 2019. – 648 с.
- Поляков К. Ю, Еремин В.А. Информатика. 10 класс. Учебник. Углубленный уровень. В 2 частях. ФГОС. – БИНОМ. Лаборатория знаний, 2019. – 560с.
- Поляков К. Ю, Еремин В.А. Информатика. 8 класс. Учебник. ФГОС. – БИНОМ. Лаборатория знаний, 2019. – 256 с.
- Поляков К. Ю, Еремин В.А. Информатика. 9 класс. Учебник. ФГОС. – БИНОМ. Лаборатория знаний, 2019. – 288 с.
- Программирование на языке Python: учебный курс.: Пер. с англ. – СПб.: ООО "Альфа-книга": 2019. – 736 с.
- Тарасов Д. Факультативный курс «Основы алгоритмизации и программирования на языке Python» 10-11 классы. Видеоуроки для учителей, родителей и учеников.

Электронные образовательные ресурсы в открытом доступе

- <http://inf1september.ru/article.php?ID=200801401>
- <https://younglinux.info/python.php>
- <https://younglinux.info/python/task/>

Календарный учебный график

№п/п	Дата по плану	Дата по факту	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
1.			Комбинированное	1	Инструктаж по ТБ. Понятие алгоритм и программа. Начальное знакомство с языком.	Аудитория Точки Роста	Опрос
2.			Комбинированное	1	Элементы языка. Структура программы. Режимы работы Операции и переменные. Типы данных	Аудитория Точки Роста	Опрос
3.			Комбинированное	1	Ввод и вывод данных на языке Python. Ввод и отладка программ в среде	Аудитория Точки Роста	Опрос
4.			Комбинированное	1	Линейные алгоритмы целочисленных данных и их реализация на Python. Решение задач по теме «Обработка целочисленных данных»	Аудитория Точки Роста	Беседа
5.			Практическое	1	Линейные алгоритмы вещественных чисел и их реализация на Python. Решение задач по теме «Обработка вещественных чисел»	Аудитория Точки Роста	Беседа
6.			Практическое	1	Логические выражения и операторы. Алгоритм «Выбор». Реализация алгоритма ветвления на Python. Примеры решения задач.	Аудитория Точки Роста	Опрос
7.			Комбини	1	Решение задач по теме «Полное и	Аудитория	Беседа

			рованное		неполное ветвление»	Точки Роста	
8.			Комбини рованное	1	Решение задач по теме «Полное и неполное ветвление»	Аудитория Точки Роста	Беседа
9.			Комбини рованное	1	Множественный выбор и его реализация с помощью вложенных ветвлений.	Аудитория Точки Роста	Беседа
10.			Комбини рованное	1	Решение задач по теме. Сложные условия. Каскадные ветвления.	Аудитория Точки Роста	Беседа
11.			Практи- ческое	1	Циклические алгоритмы.	Аудитория Точки Роста	Опрос
12.			Практи- ческое	1	Циклические алгоритмы. Инструкции управления циклом	Аудитория Точки Роста	Беседа
13.			Практи- ческое	1	Циклические алгоритмы. Инструкции управления циклом	Аудитория Точки Роста	Беседа
14.			Комбини рованное	1	Сложные циклические алгоритмы. Вложенные циклы.	Аудитория Точки Роста	Беседа
15.			Комбини рованное	1	Вложенные циклы, решение задач.	Аудитория Точки Роста	Беседа
16.			Комбини рованное	1	Вложенные циклы, решение задач.	Аудитория Точки Роста	Беседа
17.			Комбини рованное	1	Этапы решения задачи на компьютере. Последовательное конструирование алгоритма	Аудитория Точки Роста	Беседа
18.			Комбини рованное	1	Вспомогательные алгоритмы и подпрограммы. Функции	Аудитория Точки Роста	Беседа

19.			Комбини рованное	1	Вспомогательные алгоритмы и подпрограммы. Применение функций при решении задач.	Аудитория Точки Роста	Опрос
20.			Комбини рованное	1	Вспомогательные алгоритмы и подпрограммы. Применение функций при решении задач.	Аудитория Точки Роста	Опрос
21.			Комбини рованное	1	Вспомогательные алгоритмы и подпрограммы. Рекурсия.	Аудитория Точки Роста	Беседа
22.			Комбини рованное	1	Вспомогательные алгоритмы и подпрограммы. Рекурсия.	Аудитория Точки Роста	Беседа
23.			Комбини рованное	1	Структурные типы данных. Обработка списков. Линейный поиск. Добавление и удаление элементов. Копирование списков	Аудитория Точки Роста	Опрос
24.			Практи- ческое	1	Структурные типы данных. Решение задач по теме.	Аудитория Точки Роста	Беседа
25.			Комбини рованное	1	Структурные типы данных. Кортежи.	Аудитория Точки Роста	Беседа
26.			Комбини рованное	1	Структурные типы данных. Словари.	Аудитория Точки Роста	Беседа
27.			Практи- ческое	1	Символьные строки. Типовые задачи обработки строк.	Аудитория Точки Роста	Опрос
28.			Практи- ческое	1	Решение задач по теме "Символьные строки"	Аудитория Точки Роста	Беседа
29.			Практи-	1	Матрицы. Основные понятия.	Аудитория	Беседа

			ческое			Точки Роста	
30.			Практи- ческое	1	Матрицы. Типовые задачи обработки элементов.	Аудитория Точки Роста	Беседа
31.			Практи- ческое	1	Матрицы. Решение задач обработки элементов.	Аудитория Точки Роста	Опрос
32.			Практи- ческое	1	Матрицы. Решение задач обработки элементов.	Аудитория Точки Роста	Беседа
33.			Практи- ческое	1	Типы файлов. Запись информации в файл, чтение из файла	Аудитория Точки Роста	Беседа
34.			Практи- ческое	1	Изменение файлов. Решение задач	Аудитория Точки Роста	Беседа
35.			Практи- ческое	1	Защита творческих работ.	Аудитория Точки Роста	Защита итогового проекта
36.			Практи- ческое	1	Обзор пройденного материала. Подведение итогов обучения	Аудитория Точки Роста	Защита итогового проекта
Всего часов		36					