

Управление образования и молодежной политики администрации
Павловского муниципального округа Нижегородской области

Муниципальное автономное общеобразовательное
учреждение средняя школа №9 с углубленным изучением
отдельных предметов г.Павлово

Рассмотрена и принята
на педагогическом совете
Протокол №1 от 30.08.2024 г.

УТВЕРЖДАЮ
Директор школы
_____ Д.С. Соколов
« 30 » августа 2024 г.

Дополнительная общеобразовательная
общеразвивающая программа
технической направленности
*«Начала программирования на языках
C++Python»*

Возраст обучающихся: 13-18 лет
Срок реализации: 1 год

Автор-составитель: Тупицкая Татьяна
Викторовна, педагог дополнительного
образования

г. Павлово, 2024 г.

Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Начала программирования на языках C++Python» разработана в соответствии с Федеральным Законом от 29.12.2012 г. №273 – ФЗ «Об образовании в Российской Федерации» и Приказом Министерства просвещения Российской Федерации от 09.11.2018 г. № 196. «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

В наше компьютеризированное время важно, чтобы человек не только умел работать за компьютером, но и понимал, как устроены программы, с помощью которых он работает на нём.

Для изучения основных принципов программирования был выбран язык Python. Его синтаксис достаточно прост и интуитивно понятен, что понижает порог вхождения и позволяет сосредоточиться на логических и алгоритмических аспектах программирования, а не на выучивании тонкостей синтаксиса. При этом Python является очень востребованным языком; он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения.

Научившись программировать на языке Python, учащиеся получат мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность его конструкций позволит учащимся потом с лёгкостью выучить любой другой язык программирования.

Актуальность программы состоит в том, что изучение основ программирования связано с развитием целого ряда таких умений и навыков, которые носят общеинтеллектуальный характер и формирование которых – одна из приоритетных задач современной школы. Изучение программирования развивает мышление школьников, способствует формированию у них многих приемов умственной деятельности. Здесь роль информатики сродни роли математики в школьном образовании. Поэтому не

использовать действительно большие возможности программирования для развития мышления школьников, формирования многих общеучебных, общеинтеллектуальных умений и навыков было бы, наверное, неправильно.

Педагогическая значимость данной программы заключается также в том, что в ней большое внимание уделяется исследовательской деятельности, так как многие практические задачи, встающие перед учащимися, требуют грамотного и эффективного умения распределять свою работу по времени, искать свои ошибки и анализировать работу в целом. К тому же, занятия посещают и одаренные дети и дети из семей социального риска, что положительно скажется на их воспитании, развитии, саморазвитии и самореализации.

Программа направлена на реализацию технического направления.

Основные принципы реализации модульность и практическая направленность, что обеспечит вариативность обучения.

Содержание учебных модулей направлено:

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

Адресат программы. Программа предназначена для работы с детьми среднего и старшего школьного возраста (13 -18 лет). Набор учащихся в объединение свободный, независимо от национальной и половой принадлежности, социального статуса родителей (или законных представителей). Рекомендуемое количество детей в группе – 15 человек.

Цель программы: Интеллектуальное и творческое развитие учащихся на базе повышенного познавательного интереса к информатике, основанного на поиске новых познавательных ориентиров с последующей профессиональной деятельностью в различных предметных областях.

Задачи программы:

Предметные:

- изучение методов программирования на языке программирования Python;
- приобретение навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- формирование навыков использования языка методов программирования в учебной и последующей профессиональной деятельности в различных предметных областях;
- научить учащихся ориентироваться и продуктивно действовать в информационном Интернет-пространстве, используя для достижения своих целей создаваемые веб-ресурсы;
- формирование знаний о способах научно-технического мышления и деятельности, направленных на самостоятельное творческое познание и исследование информационной части сетевого пространства;
- формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- знакомство с принципами и методами функционального программирования;
- формирование знаний о принципах и методах объектно-ориентированного программирования;

Метапредметные:

- формирование у обучаемых творческого мышления, способности к самостоятельному и инициативному решению проблем;
- реализовать коммуникативные, технические и эвристические способности учащихся в ходе проектирования и конструирования сайтов.
- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- систематизации знаний по математическим основам информатики. –

развитие у обучающихся интереса к программированию.

Личностные:

- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе, приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;
- способны осуществлять рефлексивную деятельность, оценивать свои результаты, корректировать дальнейшую деятельность.

Объем и срок реализации программы, режим занятий. Программа рассчитана на 1 год обучения учащихся в возрасте 13-18 лет (7-11 классы), но в случае необходимости может быть реализована в течение более длительного срока. Время, отведенное на обучение, составляет 68 часов в год, причем практические занятия составляют большую часть программы.

Занятия проходят 1 раз в неделю, продолжительность занятия два академических часа по 40 минут с перерывом в 10 минут. Учебный год начинается 1 сентября и заканчивается 27 мая.

70% содержания планирования направлено на практическую

деятельность.

Формы организации образовательного процесса

Основной тип занятий - комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств.

Единицей учебного процесса является блок уроков (модуль). Каждый такой блок охватывает отдельную информационную технологию или её часть. Внутри блоков разбивка по времени изучения производится учителем самостоятельно, но с учётом рекомендованного календарно-тематического плана. С учётом регулярного повторения ранее изученных тем темп изучения отдельных разделов блока определяется субъективными и объективными факторами.

Каждая тема курса начинается с постановки задачи — характеристики предметной области или конкретной программы на языке Python, которую предстоит изучить. С этой целью учитель проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующих минимальному уровню планируемых результатов обучения.

Основные задания являются обязательными для выполнения всеми обучающимися в группе. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом ученики не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

В каждую тему курса интегрированы вопросы, связанные с теоретическими основами изучаемого раздела. Это позволяет учащимся получить более глубокое представление о принципах работы с информацией и способах автоматизации этой деятельности.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый ученик получил наилучший результат обучения, программой предусмотрены дополнительные задания для самостоятельной работы на домашнем компьютере. Такая форма организации обучения стимулирует интерес ученика к предмету, активность и самостоятельность учащихся, способствует объективному контролю глубины и широты знаний, повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться.

Выполнение контрольных заданий способствует активизации учебно-познавательной деятельности и ведёт к закреплению знаний, а также служит индикатором успешности образовательного процесса.

Методы организации учебного процесса

При организации занятий по программе «Начала программирования» используются формы проведения занятий с активными методами обучения:

- занятие в форме проблемно-поисковой деятельности; –
- занятие с использованием межпредметных связей;
- занятие в форме мозгового штурма;
- занятие в форме частично-поисковой деятельности.

Планируемые результаты освоения программы курса «Начала программирования»

В результате реализации программы у обучающихся будут сформированы личностные, регулятивные, познавательные и коммуникативные универсальные учебные действия.

Предметные результаты:

Учащиеся должны знать:

- правила выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей,
- знать о математических объектах информатики и об их свойствах, умение оперировать с числами в различных системах счисления, измерять количество информации, представлять информацию в структурированном виде;
- знать основы предметных понятий «информация», «алгоритм», «исполнитель», «модель» и их свойства;
- правила разработки программ в выбранной среде программирования, включая тестирование и отладку программ;
- основы несложных программ анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;

Учащиеся должны уметь:

- составлять и записывать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;
- разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу.

Метапредметные результаты:

Учащиеся должны уметь:

- самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- планировать пути решения поставленной проблемы для получения эффективного результата; понимать, что в программировании длинная программа не всегда лучшая;
- критически оценивать правильность решения учебно-исследовательской задачи;
- корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владеть основами самоконтроля, способность к принятию решений;
- создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Личностные результаты:

- у учащихся сформировано ответственное отношение к учению, способности 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;
- у учащихся развит опыт участия в социально значимых проектах, повышен уровень самооценки благодаря реализованным проектам и участия в конкурсах и конференциях различного уровня;

- сформирована коммуникативная компетентность в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности,
- сформированы ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Формы контроля и аттестации

Для полноценной реализации данной программы используются текущий и промежуточный виды контроля.

Формы текущего контроля теоретической подготовки: устный опрос, практические работы, участие в проектной деятельности.

Формы текущего контроля практической подготовки: участие в научно-практических конференциях, олимпиадах, конкурсах.

Требования к результатам выполнения исследовательского проекта:

- умение планировать и осуществлять проектную и исследовательскую деятельность;
- способность презентовать достигнутые результаты, включая умение определять приоритеты целей с учетом ценностей и жизненных планов;
- самостоятельно реализовывать, контролировать и осуществлять коррекцию своей деятельности на основе предварительного планирования;
- способность использовать доступные ресурсы для достижения целей;
- осуществлять выбор конструктивных стратегий в трудных ситуациях;
- способность создавать продукты своей деятельности, востребованные обществом, обладающие выраженными потребительскими свойствами;
- сформированность умений использовать все необходимое

многообразии информации и полученных в результате обучения знаний, умений и компетенций для целеполагания, планирования и выполнения индивидуального проекта.

**Формы и разделы текущего контроля по курсу
«Начала программирования»**

№	Разделы	Форма контроля
1.	Типы данных в программировании. Определение переменной	<i>Практические работы</i>
2.	Логические структуры	
3.	Циклы	
4.	Списки, массивы и словари	
5.	Функции	

Результаты текущего контроля анализируются педагогом дополнительного образования по следующим уровням:

- высокий уровень;
- средний уровень; –
базовый уровень.

Формы и порядок проведения промежуточной аттестации.

Для определения уровня обученности учащихся по дополнительной общеразвивающей программе используется система оценивания теоретической знаний и практической подготовки учащихся.

Промежуточная аттестация проводится самостоятельно педагогом дополнительного образования. Во время проведения промежуточной аттестации может присутствовать администрация школы. Промежуточная аттестация учащихся проводится по итогам учебного года (май).

Оценка, оформление и анализ результатов промежуточной аттестации.

Для определения уровня обученности учащихся по дополнительной общеразвивающей программе используется система оценивания подготовки учащихся. Теоретическая подготовка проверяется через выполнение **практических работ** (приложение), практическая подготовка – **защита проекта.**

По итогам года выполняется творческий проект, в рамках промежуточного контроля с 15 мая, защита которого может осуществляется по нескольким уровням.

- муниципальный – высокий уровень,
- школьный уровень - средний уровень,
- внутригрупповой уровень - базовый уровень,

Результаты участия учащихся в мероприятиях областного и выше уровней в течение учебного года оценивается педагогом как предварительная сдача промежуточной аттестации.

Оценка, оформление и анализ результатов промежуточной аттестации.

Основным контрольно-измерительным материалом является итоговый протокол, в котором фиксируется в суммарное значение теоретической и практической части прохождения промежуточной аттестации учащихся.

Протокол

результатов промежуточной аттестации, обучающихся по дополнительным общеобразовательным общеразвивающим программам МБОУ СШ №9 с углубленным изучением отдельных предметов г.Павлово

20___/20___ учебный год

Объединение: _____ Год

обучения _____ ФИО

педагога _____ Дата

проведения аттестации _____ Форма

проведения _____

Форма оценки результатов аттестации: уровень (высокий, средний, базовый)

№ п/п	Ф.И.О. учащегося	Теоретическая	Практическая подготовка	Результат
		подготовка		Уровень аттестации
1.	Иванов И.	Высокий	Высокий	высокий

Учебный план программы

Всего часов	В том числе теория	В том числе практика	Форма текущего контроля
68	32	36	<p><u>Теоретическая часть:</u> устный опрос, практические работы, участие в проектной деятельности.</p> <p><u>Практическая часть:</u> участие в научно-практических конференциях, олимпиадах, конкурсах.</p>

Календарный учебный график
дополнительной общеобразовательной общеразвивающей программы
«Начала программирования на языках C++Python»
на 2024-2025 учебный год

Комплектование групп проводится с 1 по 15 сентября 2024 года.

Продолжительность учебного года составляет 34 учебных недель. Учебные занятия в МАОУ СШ №9 с углубленным изучением отдельных предметов г. Павлово начинаются с 1 сентября 2024 г. и заканчиваются 27 мая 2025 г.


Учебные занятия проводятся во 2-ю смену (в соответствии с расписанием). Продолжительность занятий составляет 40 мин.


Каникулы: зимние каникулы с 30.12.2024 г. по 12.01.2025 г.; летние каникулы с 01.06.2025 г. по 31.08.2025 г.

В каникулярное время занятия в объединениях не проводятся. Во время каникул учащиеся могут принимать участие в мероприятиях в соответствии с планами воспитательной работы школы и классного руководителя.

Год обучения	Сентябрь				Октябрь				Ноябрь				Декабрь				Январь				Февраль				Март				Апрель				Май				Июнь	Июль	Август	Всего учебных недель/часов
	07.09	14.09	21.09	28.09	05.10	12.10	19.10	26.10	09.11	16.11	23.11	30.11	07.12	14.12	21.12	28.12	04.01	11.01	18.01	25.01	01.02	08.02	15.02	22.02	01.03	15.03	29.03	05.04	12.04	19.04	26.04	03.05	10.05	17.05	24.05	01.06 – 30.06	01.07 – 31.07	01.08 – 31.08		
1 год	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	К	К	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	П - 2	К	К	К	34/34

Условные обозначения:

Промежуточная аттестация – 

Ведение занятий по расписанию – 

Каникулярный период – 

Проведение занятий не предусмотрено расписанием – 

Рабочая программа курса

№ п/п	Название разделов, тем	Количество учебных часов			Форма контроля
		Всего	Теория	Практика	
1-2.	Интерактивный режим Python. Установка Python на компьютер.	2	1	1	Практическая работа
1 раздел. Типы данных в программировании. Определение переменной (4 часа)					
3-4.	Переменные. Стандартный ввод/вывод. Операции с целыми числами.	2	1	1	
5-6.	Операции с вещественными числами. Типы данных.	2	1	1	Практическая работа
2 раздел. Логические структуры (10 часов)					
7-8.	Логические операции, операции сравнения.	2	1	1	
9-12.	Условия: if, else, elif	4	2	2	
13-16.	Множественное ветвление	4	2	2	Практическая работа
3 раздел. Циклы - 18 часов.					
17-22.	Цикл while.	6	3	3	
23-28.	Операторы break, continue	6	3	3	
29-34.	Цикл for	6	3	3	Практическая работа
4 раздел. Списки, массивы и словари (14 часов)					
35-36.	Понятие массива и списка.	2	1	1	
37-38.	Создание списка/массива и добавление и удаление элементов.	2	1	1	
39-40.	Доступ к элементам списка/массива и их изменение	2	1	1	
41-42.	Поиск в списке/массиве	2	1	1	Практическая работа
43-44.	Сортировка списков/ массивов	2	1	1	
45-46.	Таблицы данных	2	1	1	
47-48.	Введение в словари	2	1	1	Практическая работа
5 раздел. Функции (16 часов)					
49-52.	Вызов функции	4	2	2	Практическая работа
53-56.	Параметры и аргументы функций	4	2	2	Практическая работа
57-64.	Функции с несколькими аргументами	8	4	4	Практическая работа
6 раздел. Разработка и защита проекта (4 часа)					
65-66.	Разработка проекта	2		2	Практическая работа
67-68.	Промежуточная аттестация.	2		2	Защита проекта
Итого:		68	32	36	

Содержание программы курса

Занятие 1-2. Вводное занятие. Интерактивный режим Python. Установка Python на компьютер.

Раздел 1. Знакомство с Python и средой программирования (2 ч.)

Занятие 3-4. Язык программирования Python и его характерные особенности. Структура программы на языке Python. Простейшая программа. Среды разработки. Элементы языка Python. Создание и исполнение программ в среде разработки. Оператор ввода/вывода.

Раздел 2. Типы данных в программировании.

Определение переменной (4 ч.)

Занятия 5-8. Переменные. Типы данных в языке Python. Простые типы данных. Целые и вещественные типы. Значения. Оператор присваивания. Операции, допустимые с переменными и значениями целого и вещественного типа.

Раздел 3. Логические структуры (10 ч.)

Занятия 9-16. Условный оператор. Полная и неполная формы условного оператора. Оператор выбора. Алгоритм поиска максимального / минимального элемента последовательности. Тестирование готовой программы.

Раздел 4. Циклы (18 ч.)

Занятия 17-34. Цикл. Цикл со счётчиком. Цикл с предусловием и цикл с постусловием. Генератор случайных чисел. Символьный тип данных. Использование возможностей модуля CRT для приёма и обработки сигналов клавиш.

Раздел 5. Списки, массивы и словари (14 ч.)

Занятия 35-48. Массив или список. Одномерный массив/список. Двумерный массив/список. Объявление массивов/списков. Обработка массивов/списков: поиск элемента по заданным признакам, заполнение массива/списка, вывод массива на экран. Сортировка массива/список. Сортировка выбором.

Пузырьковая сортировка.

Раздел 6. Функции (16 ч.)

Занятия 49-64. Примеры задач, использующие функции. Понятие класса как «старшего брата» функции. Алгоритм Евклида на функциях. Параметры и аргументы функций. Функции с несколькими аргументами. Функции возвращающие значения

Раздел 7. Разработка и защита проекта (4 ч.)

Занятия 65-68. Создание собственных проектов на любом из языков, с использованием пройденного материала

Условия реализации программы курса «Начала программирования»

В основу программы положен принцип воздействия, производимый педагогом на воспитанника (обучающегося), представляющий ему выработать потребность найти свое место в жизни, развить свои способности и таланты.

Важнейшим условием реализации программы является создание развивающей, здоровьесберегающей образовательной среды как комплекса комфортных, психолого-педагогических и социальных условий, необходимых для развития творческих интересов и способностей детей.

Программа курса разработана с учетом индивидуальных возрастных, психологических и физиологических особенностей обучающихся и определяет систему требований:

- **Целеполагание.** Перед обучающимися ставятся конкретные, достижимые, понятные, диагностируемые цели. Целеполагание осуществляется совместно с обучающимися исходя из сформулированной проблемы. Обучающиеся должны знать, какие конкретно знания, умения и способы деятельности они осваивают в процессе деятельности.
- **Мотивация.** Формирование интереса как к процессу учебной деятельности, так и к достижению конечного результата. Эффективными мотивами являются решение актуальной проблемы,

практическая направленность содержания.

- **Практическая значимость знаний и способов деятельности.** Педагог должен показать обучающимся возможности применения осваиваемых знаний и умений в их практической деятельности.
- **Отбор содержания.** Планируемые результаты, определенные программой, отрабатываются на практике.

Материально-технические условия реализации программы:

- компьютерный класс с 15 персональными компьютерами;
- ноутбук,
- локальная сеть,
- глобальная сеть,
- мультимедиа проектор,
- интерактивная доска.
- принтер,
- сканер,
- среда программирования Thonny.

Учебно-методическое и информационное обеспечение программы

Список литературы:

Литература для педагога:

1. К. Ю. Поляков, «Программирование. Python, C++», М.: БИНОМ. Лаборатория знаний, 2018
2. К. Ю. Поляков, Е. А. Еремин. Информатика. Углублённый уровень. Учебник для 8, 9, 10 классов. М.: БИНОМ. Лаборатория знаний,
3. Васильев А.Н., «Python на примерах. Практический курс», Наука и Техника, 2019 г.
4. С. М. Окулов. Основы программирования. М.: Бином. Лаборатория знаний, 2012.
5. Д.М. Златопольский, «Основы программирования на языке Python»,

ДМК-Пресс, 2018 г.

6. Джейсон Бриггс. «Python для детей. Самоучитель по программированию», Манн, Иванов и Фербер, 2020 г.

Литература для учащихся:

1. К. Ю. Поляков, «Программирование. Python, C++», М.: БИНОМ. Лаборатория знаний, 2018
2. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И. Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.
3. Д.М. Златопольский: Основы программирования на языке Python.: ДМК-Пресс, 2018 г.
4. Джейсон Бриггс. «Python для детей. Самоучитель по программированию», Манн, Иванов и Фербер, 2020 г.

Список рекомендуемых Интернет-ресурсов:

1. <http://phys.bspu.unibel.by/static/um/inf/prg/sem1/reshebnik/main.htm> 2. <http://pythontutor.ru/>
3. <https://sites.google.com/site/pythonpavlovo/> 4. <https://pythonworld.ru/>
5. <https://tproger.ru/tag/python/>

Перечень практических работ

1. Форматированный вывод. Вывод на экран рисунок с помощью символов 2.
Вычисления по формулам. Расчет треугольника и других фигур
 3. Работа с разрядами целого числа
 4. Составление и вычисление логических выражений 5.
Полные и неполные ветвления
 6. Сложные условия
 7. Цикл while и целые числа
 8. Цикл while и клавиатурный ввод
 9. Цикл while и генератор случайных чисел
 10. Рекуррентные соотношения
 11. Расчеты сходящихся числовых рядов с точностью eps
 12. Цикл for и клавиатурный ввод
 13. Цикл for и генератор случайных чисел
 14. Использование условной инструкции if в теле цикла while
 15. Использование условной инструкции if в теле цикла for 16.
- Вложенные циклы
17. Вложенные циклы и целые числа
 18. Работа с символами строки
 19. Обработка строк с использованием оператора цикла с параметром
 20. Обработка строк с использованием операторов цикла с условием 21.
- Функции и процедуры
22. Рекурсия
 23. Инициализация массива и вывод его на экран
 24. Использование условий для анализа элементов массива/списки 25.
- Использование условий для изменения элементов массива/списки
26. Расчет суммы, количества и среднего арифметического элементов массива, удовлетворяющих некоторому условию
 27. Поиск максимума и минимума
 28. Перестановка элементов массива/списка

29. Работа с двумя массивами/списками
30. Двумерные массивы/списки
31. Структуры и массивы/списки структур
32. Динамические массивы/списки. Передача массивов в функции
33. Текстовые файлы. Запись и чтение текстового файла
34. Поиск в текстовом файле. Работа с несколькими файлами
35. Сортировка массива/списки
36. Индивидуальные проекты

Примеры практических работ

1. Форматированный вывод. Вывод на экран рисунок с помощью символов

<p>1) Получить на экране следующее: Три числа: 103, 25, 724 Текст 'Три числа: 103, 25, 724' не использовать.</p>
<p>2) Получить на экране следующее: ○○○○○124○○○○○○13 ○○○○○○56○○○○○○355 ○○○○○587○○○○○○○8 где символом «○» обозначена пустая позиция на экране.</p>

2. Вычисления по формулам. Расчет треугольника и других фигур

<p>1) Напишите программу, которая решает следующую задачу: «N школьников делят k яблок поровну так, чтобы каждому достались только целые яблоки, остальные яблоки остаются в корзинке. Определить, сколько яблок достанется каждому школьнику и сколько яблок останется в корзинке».</p>
<p>2) Даны катеты прямоугольного треугольника. Определить его гипотенузу.</p>

3. Работа с разрядами целого числа

1) Напишите программу, в которой рассчитывается сумма цифр двузначного числа, вводимого с клавиатуры.

2) Напишите программу, в которую вводится трехзначное число и выводятся на экран его цифры. Например, при вводе числа 123 программа должна вывести:

1, 2, 3

4. Составление и вычисление логических выражений

1) Даны координаты точки на плоскости. Определить, попадает ли точка в область I (рис. 5.4). Для простоты принять, что координаты точки не равны соответствующим границам этой области.

2) Определить, войдет ли в конверт с внутренними размерами a и b мм прямоугольная открытка с размерами c и d мм. Для размещения открытки в конверте необходим зазор в 1 мм с каждой стороны.

5. Полные и неполные ветвления

1) Дано натуральное число. Определить, является ли оно двузначным.

2) Даны три целых числа. Вывести на экран те из них, которые являются четными.

6. Сложные условия

1) Известен вес боксера-любителя (в кг, в виде вещественного числа). Известно, что вес таков, что боксер может быть отнесен к одной из трех весовых категорий:

а) легкий вес – до 60 кг;

б) первый полусредний вес – до 64 кг;

в) полусредний вес – до 69 кг.

Определить, в какой категории будет выступать данный боксер.

2) Даны координаты точки, проверить какой четверти она принадлежит

7. Цикл while и целые числа

1) Дано натуральное число. Определить сумму его цифр.

2) Дано натуральное число. Определить среднее арифметическое его цифр.

8. Цикл for и клавиатурный ввод

1) Напечатать «столбиком» кубы всех целых чисел от 10 до b (значение b вводится с клавиатуры; $b \geq 10$).

2) Напечатать все нечетные числа из интервала $[10, 100]$, на используя конструкцию if

9. Использование условной инструкции if в теле цикла while

1) Напечатать те натуральные числа, квадрат которых не превышает заданного числа n .

2) Даны два натуральных числа. Разработать программу для определения наибольшего общего делителя (НОД) заданных чисел.

10. Использование условной инструкции if в теле цикла for

1) Напечатать все нечетные двузначные числа, у которых последняя цифра равна 3 или 7.

2) Напечатать все целые числа от a до b , кратные некоторому числу c .