

Муниципальное общеобразовательное учреждение Силикатненская средняя школа имени В.Г. Штыркина.

Рассмотрена на заседании
педагогического совета
Протокол № 10
от 22.06.2023



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
технической направленности
«Основы программирования»**

Возраст обучающихся: *14-16 лет*
Срок реализации: *1 год*
Уровень программы: *базовый*

Разработчик программы:
*Педагог дополнительного образования
Замалетдинова Елена Вячеславовна*

п. Силикатный, 2023 г.

Содержание

1. Комплекс основных характеристик программы	3
1.1. Пояснительная записка	3
1.2. Цель и задачи программы.....	5
1.3. Планируемые результаты освоения программы	5
1.4. Учебно-тематический план	6
1.5. Содержание учебно-тематического плана	8
2. Комплекс организационно-педагогических условий	12
2.1. Календарный учебный график	12
2.2. Формы аттестации/контроля	19
2.3. Оценочные материалы	19
2.4. Методическое обеспечение программы.....	19
2.5. Условия реализации программы.....	20
2.6. Воспитательный компонент	22
3. Список литературы	23

1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1. Пояснительная записка

Нормативно-правовое обеспечение программы:

Дополнительная образовательная общеразвивающая программа "Основы программирования»разработана в соответствии со следующими нормативно-правовыми документами:

Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации» (ст. 2, ст. 15, ст.16, ст.17, ст.75, ст. 79);

Концепция развития дополнительного образования до 203года, утвержденной распоряжением Правительства Российской Федерации от 31.03.2022 года № 678-р;

Приказ Министерства просвещения Российской Федерации от 27 июля 2022 года № 629 «Об утверждении порядка организации образовательной деятельности по дополнительным общеобразовательным программам»;

Методические рекомендации по проектированию дополнительных общеразвивающих программ № 09-3242 от 18.11.2015 года;

СП 2.4.3648-20 Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи;

Локальные акты образовательной организации:

Устав образовательной организации МОУ Силикатненская средняя школа им. В.Г. Штыркина;

Положение о разработке, структуре и порядке утверждения дополнительной общеобразовательной общеразвивающей программы в МОУ Силикатненская средняя школа им. В.Г. Штыркина;

Положение о порядке проведения входного, текущего контроля, итогового контроля освоения обучающимися дополнительных общеобразовательных общеразвивающих программ, промежуточной и итоговой аттестации обучающихся в МОУ Силикатненская средняя школа им. В.Г. Штыркина;

Направленность (профиль): техническая

Актуальность программы:

С развитием современных информационных технологий сегодня любой учащийся под руководством опытного педагога может с лёгкостью научиться программировать.

Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями. Языки программирования можно сравнить с иностранными языками, овладеть ими может каждый. Учиться программировать очень интересно. Результат программирования очень часто виден сразу. Кроме того, создание компьютерных игр и обучающих программ способствует развитию логики и

креативного мышления. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

Отличительной особенностью данной программы является: · Проектная деятельность; · Информационно-коммуникационные технологии; · Применение полученных знаний на практике; · Обсуждение проектов обучающихся; · Основные этапы системного программирования; · Основы работы с техническим заданием; · Профессиональная ориентация.

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

Новизна программы: состоит в формировании информационно-коммуникационной компетентности.

Также заключается в том, что Pascal дает более широкие возможности в области программирования, На языке Pascal можно легко и быстро создавать простые компьютерные программы.

Адресат программы:

Программа предназначена для обучения детей (подростков) в возрасте 14-16 лет.

Программа ориентирована на детей, склонных связать свою будущую жизнь с профессией программиста.

Набор производится по желанию учащихся и их родителей.

Дополнительная общеразвивающая программа «Основы программирования» составлена с учётом возрастных и психофизиологических особенностей детей младшего школьного возраста.

В школьном возрасте происходит рост стремления обучающихся к достижениям. Поэтому основным мотивом деятельности обучающегося в этом возрасте является мотив достижения успеха. Именно в этом возрасте обучающийся переживает свою уникальность и стремится к совершенству.

Уровень освоения программы: базовый

Наполняемость группы: 12 человек

Объем программы: 144 часа

Срок освоения программы: 1 год

Режим занятий: 2 раза в неделю по 2 часа (2 раза по 45 минут с перерывом 10 минут).

Форма(ы) обучения: очная, возможно с применением дистанционных образовательных технологий

Особенности организации образовательного процесса:

групповые – для всей группы при изучении общих практических и теоретических вопросов. Наполняемость группы 12 человек. Состав группы постоянный.

В ходе реализации программы применяется дифференцированный, индивидуальный подход к каждому обучающемуся.

1.2. Цель и задачи программы

Цель 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 Pascal.

Задачи программы:

Образовательные:

- обучить языку программирования Pascal и созданию программ на его основе;
- научить создавать прикладное программное обеспечение;
- расширить кругозор обучающихся в области программирования;
- научить дизайнерскому оформлению созданного ПО.

Развивающие:

- развивать память и внимание, познавательную и творческую активность;
- развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;
- развивать логическое мышление.

Воспитательные:

- прививать интерес к активному творческому самовыражению, культуре труда;
- воспитывать упорство в достижении желаемого результата;
- воспитывать эстетический вкус;
- воспитывать чувство взаимопомощи, доверия, коллективизма.

1.3. Планируемые результаты освоения программы

В результате изучения курса обучающиеся достигнут следующих результатов.

Личностные: - формирование собственного жизненного опыта значимости подготовки в области программирования в условиях развития информационного общества;

- повысят образовательный уровень по использованию средств и методов программирования;

- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.

- формирование способности обучающихся к саморазвитию и личностному самоопределению, мотивации к целенаправленной познавательной деятельности с целью приобретения профессиональных навыков в ИТ-сфере;

- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Предметные:

- формирование представления об основных изучаемых понятиях:

информация, алгоритм, модель – и их свойствах;

- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе;
- формирование умений составить и записать алгоритм для конкретного исполнителя;
- формирование знаний об алгоритмических конструкциях, логических значениях и операциях;
- изучение одного из языков программирования – PascalABC.NET.

Метапредметные:

- формирование информационно-логических умений: определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;
- овладение умениями самостоятельно планировать пути достижения целей; соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией; оценивать правильность выполнения учебной задачи; - овладение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- овладение основными универсальными умениями информационного характера: постановка и формулирование проблемы; поиск и выделение необходимой информации, применение методов информационного поиска; структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий; самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (далее ИКТ-компетентности).

1.4. Учебно-тематический план

Модуль 1

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Введение в Pascal	14	6	8	
1.1.	Язык Pascal/ Среда программирования. Особенности ввода-вывода	4	2	2	Практическое задание
1.2.	Типы данных, операции. Оператор присваивания	4	2	2	Выполнение упражнений
1.3.	Числа. Стандартные операции	6	2	4	Практическое задание
2.	Алгоритмические конструкции	22	6	16	

2.1.	Условный оператор	2	2	-	Опрос
2.2.	Цикл while. Теория чисел	10	2	8	Практическое задание
2.3.	Цикл for	10	2	8	Практическое задание
3.	Строки	20	8	12	
3.1.	Литералы строк	4	2	2	Практическое задание
3.2.	Срезы строк	4	2	2	Практическое задание
3.3.	Методы строк	12	4	8	Практическое задание
4.	Функции	4	4	-	
4.1.	Парадигма и преимущества структурного программирования	2	2	-	Опрос
4.2.	Граф и стек вызовов функций.	2	2	-	Практическое задание
Итого		60	24	36	

Модуль 2

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Функции	16	4	12	
1.1	Граф и стек вызовов функций. Области видимости переменных	2		2	Практическое задание
1.2	Прямая рекурсия	10	2	8	Практическое задание
1.3	Косвенная рекурсия	4	2	2	Решение задач повышенной трудности
2	Списки и кортежи	20	8	12	
2.1	Списки и кортежи в Pascal. Сходства и различия	2	2		Опрос
2.2	Операции со списками	10	2	8	Практическое задание
2.3.	Срезы списков	4	2	2	Практическое задание
2.4	Матрицы. Операции над матрицами.	4	2	2	Решение задач повышенной трудности
3	Словари и множества	8	4	4	
3.1.	Словари	4	2	2	Практическое задание
3.2.	Множества	4	2	2	Практическое задание
4.	Объектно-ориентированное программирование	38	4	34	
4.1.	Классы в Pascaln	2	2		Опрос
4.2.	Разработка собственного класса	6	2	4	Практическое

					задание
4.3.	Разработка и программирование собственного проекта	30		30	
5.	Заключительное занятие	2		2	
5.1	Подведение итогов. Индивидуальный проект	2		2	Защита проекта
Итого		84	20	64	

1.5. Содержание учебно-тематического плана

Модуль 1.

1. Введение в Pascal

Основные понятия: трансляция, интерпретация, компиляция, синтаксис, семантика, прагматика, переменная, динамическая типизация, служебные слова, идентификаторы, простые типы данных, приоритеты операций, литералы чисел, операция присваивания.

1.1. Язык Pascal. Среда программирования. Особенности ввода-вывода

Теория. Язык программирования Pascal. Достоинства и недостатки. Области применения. Интерактивный режим работы программы.

Практика. Установка языка программирования Pascal.

1.2. Типы данных, операции. Оператор присваивания

Теория. Ввод и вывод числовой информации.

Практика. Тренировочное задание на ввод и вывод числовой информации.

1.3. Числа. Стандартные операции

Теория. Стандартные операции с целыми и действительными числами. Силь программирования .

Практика. Решение простых задач в интерактивном режиме.

2. Алгоритмические инструкции

Основные понятия: логический тип данных, логические операции (and, or, not, A), условный оператор, условное и альтернативное исполнение алгоритма, операторы сравнения, вложенность операторов, оператор цикла, переменная-флаг, генерация псевдослучайной последовательности, инструкции break, continue и pass.

2.1. Условный оператор

Теория. Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.

2.2. Цикл while

Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Решение задач на анализ чисел потока и целочисленной арифметики.

2.3. Цикл for

Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности.

3. Строки

Основные понятия: символ, строка, литерал, таблицы кодов ASCII, UTF-8, отладка кода, неизменяемый объект, формат вывода строки, экранированные escape-последовательности, положительная и отрицательная нумерация символов в строке, срез, конкатенация, длина строки.

3.1. Литералы строк

Теория. Понятие «литералы строк».

Практика. Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.

3.2. Срезы строк

Теория. Форматирование строки.

Практика. Преобразование строки. Решение упражнений.

3.3. Методы строк

Теория. Методы работы со строкой.

Практика. Применение методов строки. Решение задач.

4. Функции

Основные понятия: подпрограмма, функция, процедура, рекурсия, глубина рекурсии, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа с памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация, lambda-функции.

4.1. Парадигма и преимущества структурного программирования

Теория. Обзор парадигм программирования. Особенности применения языков программирования.

4.2. Граф и стек вызовов функций. Области видимости переменных

Теория. Применение стек и граф вызовов функций.

Практика. Выполнение тренировочных упражнений.

Модуль 2.

1. Функции

Основные понятия: подпрограмма, функция, процедура, рекурсия, глубина рекурсии, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа с памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация, lambda-функции.

1.1. Граф и стек вызовов функций. Области видимости переменных

Теория. Применение стек и граф вызовов функций.

Практика. Выполнение тренировочных упражнений.

1.2. Прямая рекурсия

Теория. Понятие прямой рекурсии и ее применение.

Практика. Выполнение тренировочных упражнений.

1.3. Косвенная рекурсия

Теория. Понятие косвенной рекурсии и ее применение.

Практика. Выполнение тренировочных упражнений.

Практические занятия к темам 4.2., 4.3., 4.4.: Нахождение суммы чисел. Вычисление степени. Числа Фибоначчи. Ханойские башни

Использование библиотеки математических функций. Решение задач повышенной трудности.

2. Списки и кортежи

Основные понятия: список, кортеж, элемент списка и кортежа, индекс, срез списка, матрица, многомерный список, сортировка, сложность алгоритма, устойчивость сортировки, квадратичная, быстрая, синхронная, поразрядная сортировки списка, случайное перемешивание.

2.1. Списки и кортежи в Python. Сходства и различия

Теория. Представление списка и кортежа в памяти компьютера, сходства и различия.

2.2. Операции со списками

Теория. Способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле).

Практика. Решение задач на ввод-вывод элементов одномерного и многомерного списка.

2.3. Срезы списков

Теория. Методы работы со списком и кортежем. Методы сортировки списка.

Практика. Решение задач на ввод-вывод элементов одномерного и многомерного списка и кортежа, преобразование, поиск, замену, подсчет.

2.4. Матрицы. Операции над матрицами

Теория. Вычисление сложности алгоритма. Многомерные списки.

Практика. Решение задач повышенной трудности.

3 Словари и множества

Основные понятия: словарь, множество, ключ, кодирование.

3.1. Словари

Теория. Понятие словаря. Способы создания. Словарь, преимущества и недостатки, методы работы со словарем. Словари со смешанными значениями. Кодирование и декодирование текста.

Практика. Решение задач на заполнение, преобразование, поиск, замену, подсчет, вывод элементов словаря.

3.2. Множества

Теория. Понятие множества. Создание множеств. Множество, преимущества и недостатки, методы работы с множеством.

Практика. Решение задач повышенной трудности.

4 Объектно-ориентированное программирование (ООП)

Основные понятия: ООП, класс, метод INIT, экземпляр, наследование, полиморфизм, исключения, виджет, интерфейс, событие, техническое задание, проект, проектная деятельность, виды проектов.

4.1. Классы в Pascal

Теория. Понятия «класс», «метод INIT», «экземпляр», «наследование», «полиморфизм», «исключения», «виджет», «интерфейс», «интерфейс», «событие».

4.2. Разработка собственного класса

Теория. Принципы разработки собственного класса. Обработка и генерация исключений. Виджет, методы виджета. Графическая библиотека tkinter, класс Tk. Системные методы. События.

Практика. Создание собственного класса.

4.3. Разработка и программирование собственного проекта

Практика. Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.

5. Подведение итогов

5.1. Защита индивидуального проекта.

2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

2.1. Календарный учебный график

Место проведения: МОУ Силикатненская СШ им. В.Г. Штыркина

Время проведения занятий: 13:00-17:00

Год обучения: 2023-2024

Количество учебных недель: 36 недель.

Количество учебных дней: 190 дней.

Сроки учебных периодов: 1 полугодие – 01.09.2023-29.12.2023

2 полугодие –15.02.2024-31.05.2024

№ п/п	Название раздела, тема	Количество часов			Формы аттестации / контроля
		Всего	Теория	Практика	
1	Вводное занятие. Инструктаж по ТБ. Язык программирования. Достоинства и недостатки.	2	2		Педагогическое наблюдение, беседа
2	Области применения. Интерактивный режим работы программы.	2	1	1	Педагогическое наблюдение, беседа
3	Установка языка программирования	2		2	Практическая работа
4	Особенности ввода-вывода. Входной контроль.	2	1	1	Педагогическое наблюдение, беседа
5	Ввод и вывод числовой информации.	2	2		Педагогическое наблюдение, беседа
6	Тренировочное задание на ввод и вывод числовой информации.	2		2	Тест
7	Стандартные операции с целыми и действительными числами. Стиль программирования.	2	2		Педагогическое наблюдение, беседа
8	Решение простых задач в интерактивном режиме.	2	2		Педагогическое наблюдение, беседа
9	Решение простых задач в интерактивном режиме.	2	2		Педагогическое наблюдение, беседа
10	Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.	2	2		Педагогическое наблюдение, беседа

11	Цикл while. Теория чисел. Фильтрация потока чисел.	2		2	Практическая работа
12	Решение задач на анализ чисел потока и целочисленной арифметики (НОД и НОК)	2		2	Опрос
13	Решение задач на анализ чисел потока и целочисленной арифметики (проверка числа на простоту)	2		2	Педагогическое наблюдение, беседа
14	Решение задач на анализ чисел потока и целочисленной арифметики (Алгоритм Евклида)	2		2	Практическая работа
15	Решение задач на анализ чисел потока и целочисленной арифметики (Нахождение максимума и минимума)	2		2	Практическая работа
16	Цикл for. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.	2	2		Педагогическое наблюдение, беседа
17	Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности (НОД и НОК).	2		2	Практическая работа
18	Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности (Проверка числа на простоту).	2		2	Практическая работа
19	Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности (Алгоритм Евклида).	2		2	Практическая работа
20	Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности (Нахождение	2		2	Практическая работа

	максимума и минимума).				
21	Понятие «литералы строк».	2	2		Педагогическое наблюдение, беседа
22	Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.	2		2	Практическая работа
23	Форматирование строки.	2	2		Педагогическое наблюдение, беседа
24	Преобразование строки. Решение упражнений.	2		2	Практическая работа
25	Методы работы со строкой.	2	2		Педагогическое наблюдение, беседа
26	Методы работы со строкой. Решение задач.	2		2	Практическая работа
27	Применение методов строки.	2	2		Педагогическое наблюдение, беседа
28	Применение методов строки. Решение задач.	2		2	Практическая работа
29	Применение методов строки. Решение задач.	2		2	Практическая работа
30	Применение методов строки. Решение задач.	2		2	Практическая работа
31	Обзор парадигм программирования. Особенности применения языков программирования.	2	2		Педагогическое наблюдение, беседа
32	Применение стек и граф вызовов функций.	2	2		Педагогическое наблюдение, беседа
33	Выполнение тренировочных упражнений.	2		2	Опрос
34	Понятие прямой рекурсии и ее применение.	2	2		Педагогическое наблюдение, беседа
35	Выполнение тренировочных упражнений.	2		2	Практическая работа
36	Выполнение тренировочных упражнений.	2		2	Практическая работа
37	Выполнение тренировочных упражнений.	2		2	Практическая работа
38	Выполнение тренировочных упражнений.	2		2	Практическая работа
39	Понятие косвенной рекурсии и ее применение.	2	2		Педагогическое наблюдение, беседа

40	Выполнение тренировочных упражнений.	2		2	Практическая работа
41	Представление списка и кортежа в памяти компьютера, сходства и различия.	2	2		Педагогическое наблюдение, беседа
42	Способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле).	2	2		Педагогическое наблюдение, беседа
43	Решение задач на ввод-вывод элементов одномерного и многомерного списка.	2		2	Опрос
44	Решение задач на ввод-вывод элементов одномерного и многомерного списка.	2		2	Опрос
45	Решение задач на ввод-вывод элементов одномерного и многомерного списка.	2		2	Опрос
46	Методы работы со списком и кортежем. Методы сортировки списка	2	2		Педагогическое наблюдение, беседа
47	Решение задач на ввод-вывод элементов одномерного и многомерного списка и кортежа, преобразование, поиск, замену, подсчет.	2		2	Практическая работа
48	Вычисление сложности алгоритма. Многомерные списки.	2	2		Педагогическое наблюдение, беседа
49	Решение задач повышенной трудности.	2		2	Практическая работа
50	Понятие словаря. Способы создания. Словарь, преимущества и недостатки, методы работы со словарем. Словари со смешанными значениями. Кодирование и декодирование текста.	2	2		Педагогическое наблюдение, беседа
51	Решение задач на заполнение, преобразование, поиск, замену, подсчет, вывод	2		2	Практическая работа

	элементов словаря.				
52	Понятие множества. Создание множеств. Множество, преимущества и недостатки, методы работы с множеством.	2	2		Педагогическое наблюдение, беседа
53	Решение задач повышенной трудности.	2		2	Практическая работа
54	Понятия «класс», «метод INIT», «экземпляр», «наследование», «полиморфизм», «исключения», «виджет», «интерфейс», «событие».	2	2		Педагогическое наблюдение, беседа
55	Принципы разработки собственного класса. Обработка и генерация исключений. Виджет, методы виджета. Графическая библиотека tkinter, класс Tk. Системные методы. События.	2	2		Практическая работа
56	Создание собственного класса.	2		2	Опрос
57	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Педагогическое наблюдение, беседа
58	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
59	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
60	Выбор вида и темы проекта. Составление технического задания.	2		2	Практическая работа

	Программирование. Разработка технической документации и презентации проекта.				
61	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
62	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
63	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
64	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
65	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
66	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
67	Выбор вида и темы проекта. Составление технического задания. Программирование.	2		2	Практическая работа

	Разработка технической документации и презентации проекта.				
68	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
69	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
70	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
71	Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.	2		2	Практическая работа
72	Защита индивидуального проекта.	2		2	Практическая работа
	Итого:	144	48	96	

2.2. Формы аттестации/контроля

Формы аттестации/контроля для выявления предметных и метапредметных результатов:

тестирование, практическая работа, творческая работа, творческий проект,

Формы аттестации/контроля формы для выявления личностных качеств:

наблюдение, беседа, опросы, анкетирование,

Особенности организации аттестации/контроля:

Результаты освоения программы отслеживаются по итогам опросов, выполнения практических заданий. Результаты обучения по программе выявляются по итогам проведения олимпиад, соревнований по программированию, защиты проекта.

2.3. Оценочные материалы

В процессе реализации программы предусмотрены следующие виды контроля:

- входной контроль проводится с целью определения уровня знаний учащихся (Приложение № 1);
- промежуточный контроль проводится регулярно на занятиях с целью определения степени усвоения материала в форме опроса, решения задач и практических заданий;
- итоговый контроль – защита проекта.

2.4. Методическое обеспечение программы

Методические материалы:

Программа разработана на основе концептуальных положений Стратегии инновационного развития Российской Федерации на период до 2020 года.

Методики и технологии:

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

Технологии:

1. Технология проблемного диалога. Учащимся не только сообщаются готовые знания, но и организуется такая их деятельность, в процессе которой они сами делают «открытия», узнают что-то новое и используют полученные знания и умения для решения жизненных задач.
2. Технология коллективного взаимообучения («организованный диалог», «сочетательный диалог», «коллективный способ обучения (КСО)», «работа

учащихся в парах сменного состава») позволяет плодотворно развивать у обучаемых самостоятельность и коммуникативные умения.

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

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

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

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

Краткое описание работы с методическими материалами:

Во время занятий применяются групповые и индивидуальные занятия, в свою очередь подразделяются на теоретические и практические занятия.

2.5. Условия реализации программы

Для эффективной реализации настоящей программы необходимы определённые условия:

наличие помещения для учебных занятий, рассчитанного на 12 человек и отвечающего правилам СанПин;

наличие ученических столов и стульев, соответствующих возрастным особенностям обучающихся;

шкафы стеллажи для оборудования, а также разрабатываемых и готовых прототипов проекта;

наличие необходимого оборудования согласно списку;

наличие учебно-методической базы: качественные иллюстрированные определители животных и растений, научная и справочная литература, наглядный материал, раздаточный материал, методическая литература.

Материально-техническое обеспечение:

- столы для компьютера;
- компьютерные стулья;
- шкафы для дидактических материалов, пособий;
- специальная и научно-популярная литература для педагога и учащихся;

- канцтовары;

Информационное обеспечение:

- персональный компьютер (на каждого участника);
- мультимедийный проектор;
- видеоматериалы разной тематики по программе;
- оргтехника;
- выход в сеть Internet;

Аппаратное обеспечение:

- Процессор не ниже Core2 Duo;
- Объем оперативной памяти не ниже 4 Гб DDR3;
- Дисковое пространство на менее 128 Гб;
- Монитор диагональю на мене 19”;

Программное обеспечение:

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор;
- FoxitReader или другой просмотрщик PDF файлов;
- WinRAR;
- Пакет офисных программ;
- Любой браузер для интернет серфинга.

Дистанционные образовательные технологии

Отличительная особенность дополнительной общеобразовательной программы заключается в случае необходимости перейти на использование дистанционных образовательных технологий при разработке комплекса методов и приемов, направленных на развитие навыков составления презентаций. Каждый обучающийся реализует индивидуальный проект в результате освоения программы. Продукт, полученный в результате освоения программы, имеет прикладной характер и может быть использован по необходимости.

Дистанционные образовательные технологии, реализуются в основном с применением информационно-телекоммуникационных сетей при опосредованном (на расстоянии) взаимодействии обучающихся и педагогических работников.

В рамках онлайн занятий посредством платформ: Zoom, Youtube, Skype, и другие, педагог предоставляет теоретический материал по теме.

В офлайн режиме посредством социальных сетей и мессенджеров обучающимся передается видео, презентационный материал с инструкцией выполнения заданий, мастер-классы и другое.

Сетевое взаимодействие

В п. Силикатный создан Центр сетевого взаимодействия, что позволяет вести занятия в сотрудничестве с сетевыми партнерами:

МАУ «Вдохновение»;

Администрация МО «Силикатненское городское поселение»

Силикатненская модельная библиотека

УлГТУ

ДШИ им. Неклюдова
 ДЮСШ
 РДШ
 Юнармия

•Отдел по молодежной политике при Администрации МО «Сенгилеевский район»

•родительская общественность.

Все формы и виды совместной деятельности участников социальных партнёров выстраиваются на безвозмездной договорной основе, а также разнопредметных мероприятий.

Кадровое обеспечение программы

Образовательный процесс по модулям обеспечивается педагогом дополнительного образования, имеющим базовое образование, соответствующее профилю модулю или опыту.

К образовательному процессу по модулям также привлечены преподаватели из числа организаций сетевого взаимодействия.

2.6. Воспитательный компонент

Цель воспитательной работы

К воспитательным целям можно отнести: раскрытие и развитие индивидуальных способностей каждого ученика; воспитание ответственности, культуры поведения и общения; развитие творческих способностей у школьников на уроках и внеурочных занятиях; нацеливание детей на успех посредством сотрудничества, личного старания, труда, терпения и настойчивости.

Задачи воспитательной работы

Основные задачи воспитательной деятельности:

- выявлять и развивать индивидуальные особенности, личностный и творческий потенциал каждого ребенка;
- создавать благоприятную обстановку для интеллектуального, нравственного, психологического, духовного и физического развития детей, позволяющую им самоопределиваться во взрослой жизни, уважать традиции и историческое прошлое своей страны, культуру своего и других народов, уважать права и свободы личности;
- развивать интерес к учебе, воспитывать добросовестное к ней отношение и сознательную дисциплину;

Приоритетные направления воспитательной деятельности

воспитание положительного отношения к труду и творчеству, культурологическое и эстетическое воспитание, профориентационное воспитание

Формы воспитательной работы

беседа, дискуссия, акция, деловая игра,

Методы воспитательной работы

рассказ, беседа, создание воспитывающих ситуаций, соревнование,

наблюдение,

Планируемые результаты воспитательной работы

Овладение основными трудовыми умениями и навыками в рамках программы "Основы программирования".

Календарный план воспитательной работы

№ п/п	Название мероприятия	Задачи	Форма проведения	Сроки проведения
1	Онлайн конкурс по «Основам программирования»	Развитие интереса детей к научно-исследовательскому творчеству	Онлайн	В течении реализации программы

3. Список литературы

1. Великович Л. С., Цветкова М. С. Программирование для начинающих. – М.: БИНОМ.
2. Лаборатория знаний, 2011 2. Информатика. Методическое пособие для 7-9 классов. Босова Л.Л., Босова А.Ю. – М.: БИНОМ. Лаборатория знаний, 2015.
3. Лесневский А.С. Объектно-ориентированное программирование для начинающих. – М.: Бином. Лаборатория знаний, 2009
4. Программы внеурочной деятельности для основной школы. 7-9 классы. Цветкова М.С., Богомолова О.Б.– М.: БИНОМ. Лаборатория знаний, 2013
5. Семакин И.Г., Шестаков А.П. Основы алгоритмизации и программирования: учебник для сред. проф. образования. – М.: Издательский центр «Академия», 2010
6. Тишин В. И. Программирование на Паскале: практикум. – М.: БИНОМ. Лаборатория знаний, 2013.
7. Учебное пособие для 9 класса общеобразовательных учебных заведений (Издательский дом «Бином», М., 2016)
8. Быкадорова Е. А., Синявская О. Н. Основы программирования информационного контента [Электронный ресурс]:. - Санкт-Петербург: Лань, 2020. - 72 с. – Режим доступа: <https://e.lanbook.com/book/133921>
9. Основы программирования [Электронный ресурс]. - Сочи: СГУ, 2019. - 52 с. – Режим доступа: <https://e.lanbook.com/book/147661>
10. Тракимус Ю. В., Хищенко В. П. Основы программирования [Электронный ресурс]:учебное пособие. - Новосибирск: НГТУ, 2020. - 66 с. – Режим доступа: <https://e.lanbook.com/book/152224>
11. Сильвашко С. А. Основы программирования [Электронный ресурс]:учебное пособие для обучающихся - Оренбург: ОГУ, 2019. - 126 с. – Режим доступа: <https://e.lanbook.com/book/160013>

12. Кувшинов Д. Р. Основы программирования [Электронный ресурс]: Учебное пособие для вузов. - Москва: Юрайт, 2020. - 104 с – Режим доступа: <https://urait.ru/bcode/454667>

13.6. Черпаков И. В. Основы программирования [Электронный ресурс]: Учебник и практику.

Входящий контроль

Низкий уровень: 0-70% выполненных заданий;

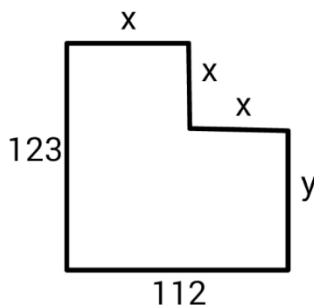
Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Для учащихся – 14-17 лет

Фамилия, Имя	
---------------------	--

15. Из прямоугольника вырезали квадрат со стороной x , получилась фигура как на рисунке. Чему равна сумма цифр y ?



16. В алфавите племени мумба-юмба 32 буквы. Любое слово в языке этого племени состоит из пяти букв и должно одинаково читаться справа налево и слева направо, при этом первые две буквы слова обязательно различаются, а третья совпадает с пятой.

Каково максимальное количество слов в этом языке?

17. Катя наклеила на рулет тонкие поперечные кольца трёх разных цветов. Если разрезать по серым кольцам, получится 25 кусков рулета, если по малиновым — 47 кусков, а если по зеленым — 31 кусок.

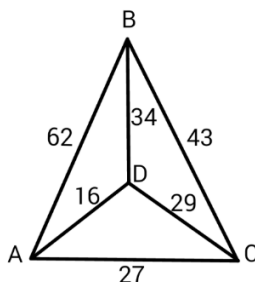
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

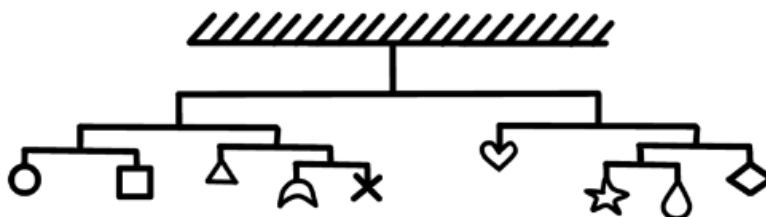
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

18. На рисунке показано расположение городов А, В, С и D и расстояния между ними. Турист выходит из города В и собирается посетить остальные города, побывав в каждом по разу.

Какова наименьшая возможная длина маршрута, если он хочет закончить свой путь в том же городе?



19. Фигурки, общей массой 432 грамма, при помощи невесомых нитей и планок собрали в конструкцию, изображённую на рисунке. Оказалось, что все её части находятся в равновесии. Сколько весит ромб?



20. Люди переезжают в города, за год численность людей удваивается. Если люди заселят весь город за 12 лет, то сколько лет понадобится, чтобы занять лишь половину города?

21. Сравните пары слов. Сколько среди них полностью идентичных?

O/Sanmarco

O/Samnarco

Ф.Wagonerrte

Ф.Wagonertre

A.S.Schmetterling

A.S.Schnetterling

N.V.Murfreesboroque

N.V.Munfreesboroque

P.S.Splendoursec

P.S.Sqlendoursec

Семь человек выясняли, какой сегодня день недели.

Первый сказал: «Послезавтра – воскресенье».

Второй: «Вчера был понедельник».

Третий: «Завтра будет суббота».

Четвертый: «Завтра будет среда».

Пятый: «Вчера был четверг».

Шестой: «Позавчера было воскресенье».

Седьмой: «Позавчера была среда».

Какой сегодня день недели, если трое ошибаются?

22. Вам предложены несколько высказываний и следствие из них (выделено жирным).

Согласны ли Вы с этим следствием?

1. Все клёны — растения.

2. Некоторые растения быстро желтеют.

Значит, некоторые клёны быстро желтеют.

	Д
	а
	Н
	ет

23. Гусеница прогрызает яблоко диаметром 6 сантиметров насквозь за 16 секунд, вылезая снаружи полностью.

Известно, что середину яблока она начинает грызть уже через 6 секунд после начала пути.

Какова длина гусеницы в сантиметрах?

Итоговый контроль
Диагностическая карта

/п	Имя	Входящая Диагностика	Промежуточ- ная диагностика	Итоговая диагностика
+				
	Итого в %	Н С В		Н С В

Результативность отслеживается с помощью карты наблюдений, анализа участия детей в совместной продуктивной деятельности, разработки и защите творческих проектов. Усвоение программы возможно по 3-м уровням: низкий (Н), средний (С), высокий (В).

Низкий уровень

Учащиеся должны знать/ понимать:

- технику безопасного поведения во время занятий;
- правила поведения в общественных местах,
- понятие программы ; общую структуру программы;
- основные типы данных;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- основные циклы с условием;
- основные правила записи циклов условием;
- формат записи цикла с параметром;
- понятие функции;
- основные принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- операции со строками;
- способ описания списка;
- способ описания кортежа;
- способ описания словаря;
- основные операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.

- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ.

Учащиеся должны уметь:

- понимать учебную задачу, сохранять ее содержание в процессе ее выполнения под руководством педагога;
- работать в паре, малой группе;
- выполнить установку программы под руководством педагога;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе под руководством педагога;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием под руководством педагога;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи по наводящим вопросам педагога;
- создавать и использовать основные функции;
- описывать и соединять строки;
- находить подстроку в строке с помощью педагога;
- находить количество слов в строке;
- вводить и выводить элементы списка под руководством педагога;
- приводить примеры использования вложенных списков (матриц) по наводящим вопросам педагога;
- описывать множества под руководством педагога;
- определять принадлежность элемента множеству по наводящим вопросам;
- определять вид ошибок и находить ошибки в программе под руководством педагога;
- составлять элементарные алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке под руководством педагога.

Средний уровень

Учащиеся должны знать/ понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности;
- основные приемы взаимодействия в группе сверстников;
- понятие программы;
- структуру программы;
- режимы работы.
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;

- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- циклы с условием и их виды;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;
- основные способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- основные шаги работы над проектом, его презентации.

Учащиеся должны уметь:

- уважительно относиться к преподавателям и сверстникам;
- применять некоторые приемы логического (абстрактное) мышления;
- концентрировать внимание на одном или двух объектах;
- понимать причины успеха/неуспеха с помощью анализа педагога;
- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использования цикла с

параметром для решения поставленной задачи;

- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.
- определять вид ошибок и находить ошибки в программе.
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ;
- отлаживать и тестировать программы;
- понимать и сохранять цели и задачи учебной деятельности в ходе занятия;
- планировать свою деятельность с помощью взрослого;
- сотрудничать с взрослыми и сверстниками в процессе выполнения учебной задачи;
- понимать причины успеха/неуспеха учебной деятельности под руководством взрослого;
- делать выводы в ходе полученных заданий; выстраивать логические цепи рассуждений под руководством педагога;
- выражать творческие идеи, разработать творческий проект на основе образца;
- конструктивно взаимодействовать в составе группы в ходе работы над проектом.

Высокий уровень

Учащиеся должны знать/ понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности, правила поведения во время чрезвычайных происшествий;
- приемы конструктивного взаимодействия в группе сверстников;
- понятие программы;
- структуру программы;
- режимы работы.

- Общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы `or`, `and`, `not`;
- циклы с условием и их виды;
- правила записи циклов условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- правила и этапы работы над проектом;
- приемы успешной презентации проекта.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной

задачи;

- использовать цикл с условием;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.
- определять вид ошибок и находить ошибки в программе.
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ;
- отлаживать и тестировать программы;
- понимать учебную задачу, анализировать достижение результата;
- делать выводы в ходе полученных заданий; самостоятельно выстраивать логические цепи рассуждений;
- понимать причины успеха/неуспеха, конструктивно принимать причины неуспеха, исправлять ошибки с минимальным участием педагога;
- планировать свою деятельность, находить оригинальные способы выполнения поставленной творческой задачи;
- создать проектировочную команду и организовать ее деятельность;
- разработать творческий проект по предложенной или самостоятельно выбранной теме в составе творческой группы;
- самостоятельно подготовить презентацию и защитить проект.

вопросы для промежуточного контроля по усвоению материала
(Промежуточный контроль проводится в последнюю неделю декабря)

Составьте выражение для вычисления в интерпретаторе и вставьте в поле ответа результат вычисления:

11111·11111111

— **произведение чисел 11111 (5 единиц) и 11111111 (7 единиц)**

Запишите число **1.2345e3** в виде десятичной дроби.

<p>Составьте и запишите выражение для вычисления: 2014.0^{14}(Возвестив 14 степень)</p> <p>Обратите внимание на запись числа: это вещественное число.</p>
<p>Приведите к целому типу число 2.99</p>
<p>Расставьте скобки в выражении a and b or not a and not b</p> <p>в соответствии с порядком вычисления выражения (приоритетом операций). Всего потребуется 5 пар скобок (внешние скобки входят в их число).</p>
<p>Найдите результат выражения для заданных значений <i>a</i> и <i>b</i> Учитывайте регистр символов при ответе. a = True b = False a and b or not a and not b</p>
<p>Отметьте выражения, значения которых равны True:</p> <ul style="list-style-type: none"> • "239" < "30" and 239 < 30 • "239" < "30" and 239 > 30 • "239" > "30" and 239 < 30 • "239" > "30" and 239 > 30
<p>Укажите результат выражения: "123" + "42"</p>
<p>Какое значение будет у переменной <i>i</i> после выполнения фрагмента программы?</p> <pre> i = 0 while i <= 10: i = i + 1 if i > 7: i = i + 2 </pre>
<p>Сколько итераций цикла будет выполнено в этом фрагменте программы?</p> <pre> i = 0 while i <= 10: i = i + 1 if i > 7: i = i + 2 </pre>
<p>Сколько всего знаков * будет выведено после исполнения фрагмента программы:</p> <pre> i = 0 while i < 5: print('*') if i % 2 == 0: print('**') if i > 2: print('***') i = i + 1 </pre>
<p>Определите, какое значение будет иметь переменная <i>i</i> после выполнения следующего фрагмента программы:</p> <pre> i = 0 s = 0 while i < 10: i = i + 1 s = s + i if s > 15: break i = i + 1 </pre>
<p>Определите, какое значение будет иметь переменная <i>i</i> после выполнения следующего</p>

фрагмента программы:

```

i = 0
s = 0
while i < 10:
  i = i + 1
  s = s + i
if s > 15:
  continue
  i = i + 1

```

Задачи для промежуточного контроля по усвоению материала

Напишите простой калькулятор, который считывает с пользовательского ввода три строки: первое число, второе число и операцию, после чего применяет операцию к введённым числам ("первое число" "операция" "второе число") и выводит результат на экран. Поддерживаемые операции: +, -, /, *, mod, pow, div, где mod — это взятие остатка от деления, pow — возведение в степень, div — целочисленное деление. Если выполняется деление и второе число равно 0, необходимо выводить строку "Деление на 0!".

Обратите внимание, что на вход программе приходят вещественные числа.

Жители страны Малевии часто экспериментируют с планировкой комнат. Комнаты бывают треугольные, прямоугольные и круглые. Чтобы быстро вычислять жилплощадь, требуется написать программу, на вход которой подаётся тип фигуры комнаты и соответствующие параметры, которая бы выводила площадь получившейся комнаты. Для числа π в стране Малевии используют значение 3.14.

Формат ввода, который используют Малевийцы:

Треугольник

a

b

c

где a, b и c — длины сторон треугольника

прямоугольник

a

b

где a и b — длины сторон прямоугольника

круг

r

где r — радиус окружности

Напишите программу, которая получает на вход три целых числа, по одному числу в строке, и выводит на консоль в три строки сначала максимальное, потом минимальное, после чего оставшееся число.

На ввод могут подаваться и повторяющиеся числа.

Паша очень любит кататься на общественном транспорте, а получая билет, сразу проверяет, счастливый ли ему попался. Билет считается счастливым, если сумма первых трех цифр совпадает с суммой последних трех цифр номера билета.

Однако Паша очень плохо считает в уме, поэтому попросил вас написать программу, которая проверит равенство сумм и выведет "Счастливый", если суммы совпадают, и "Обычный", если суммы различны.

На вход программе подаётся строка из шести цифр. (Пример:123321)

Выводить нужно только слово "Счастливый" или "Обычный", с большой буквы.

Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и сразу после этого выводит сумму квадратов всех считанных чисел.

Гарантируется, что в какой-то момент сумма введённых чисел окажется равной 0, после этого считывание продолжать не нужно.

В примере мы считываем числа 1, -3, 5, -6, -10, 13; в этот момент замечаем, что сумма этих чисел равна нулю и выводим сумму их квадратов, не обращая внимания на то, что остались ещё не прочитанные значения.

Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 5 5 5 5 ... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число n — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.

Например, если $n = 7$, то программа должна вывести 1 2 2 3 3 3 4.

Выведите таблицу размером $n \times n$, заполненную числами от 1 до n^2 по спирали, выходящей из левого верхнего угла и закрученной по часовой стрелке, как показано в примере (здесь $n=5$)

Информация для карточки в Навигаторе

Полное название: Дополнительная образовательная общеразвивающая программа "Основы программирования»

Публичное название: Основы программирования

Краткое описание:

Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Обучающиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные программы.