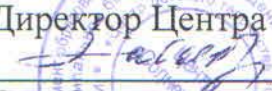


Муниципальное бюджетное учреждение дополнительного образования
«Центр детского творчества «Солнечный»

УТВЕРЖДАЮ:
Директор Центра «Солнечный»


С.В. Завьялова

Принята на заседании
Педагогического совета
Протокол № 3 от 31.05.2022

Дополнительная общеобразовательная общеразвивающая программа

«Системное программирование»

техническая направленность

Возраст учащихся: 14-17 лет
Срок реализации: 3 года
Автор: Аргов Д.И.,
педагог дополнительного образования

Рыбинск, 2022

СОДЕРЖАНИЕ

1.	Комплекс основных характеристик программы.....	3
1.1.	Пояснительная записка.....	3
1.2.	Учебно-тематический план.....	5
1.3.	Содержание программы.....	6
1.4.	Планируемые результаты.....	16
2.	Комплекс организационно-педагогических условий.....	17
2.1.	Календарный учебный график.....	17
2.2.	Методическое обеспечение программы.....	18
2.3.	Ресурсное обеспечение программы.....	21
2.4.	Оценочные материалы.....	21
2.5.	Воспитательный аспект.....	27
3.	Список источников.....	30

1. Комплекс основных характеристик программы

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Системное программирование» разработана согласно требованиям следующих нормативных документов:

– Федеральный Закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ.

– Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28 «Об утверждении СанПиН 2.4.4.3648-20 «Санитарно-эпидемиологические требования к организации воспитания и обучения, отдыха и оздоровления детей и молодежи».

– Приказ Министерства просвещения РФ от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

– Письмо Минобрнауки России № 09-3242 от 18.11.2015 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»).

– Устав и соответствующие локальные акты муниципального бюджетного учреждения дополнительного образования «Центр детского творчества «Солнечный».

Творческое объединение «Системное программирование» образовалось в 1993 году на базе компьютерного класса персональных ЭВМ "Ириша".

В настоящее время во всем мире, в том числе и в России, наметилась тенденция внедрения в науку и производство сложной электронной вычислительной техники. Персональные компьютеры проникли во все области жизни человека, даже в те, в которых раньше об этом и не помышляли. В результате этого процесса возникла проблема обеспечения населения России второй компьютерной грамотности.

В творческом объединении могут заниматься дети 8-11^x классов. Однако наиболее перспективными для обучения являются школьники 9^x классов. Это связано с их более высоким интеллектуальным и психологическим развитием, более глубокими знаниями в области математики. Обучение осуществляется в течение трех лет. 1,2,3, год обучения – 4 часа в неделю, 144 часа в год.

Программа ТО «Системное программирование» имеет техническую направленность.

Особенностями в работе нашего творческого объединения является получение учащимися глубокой теоретической базы на первом году обучения, что позволяет ему использовать эти знания, как при дальнейшем обучении, так и в школе.

Программа обучения рассчитана на три года. Занятия первого года, второго и третьего года обучения проводятся три раза в неделю по 1-2 часа.

Программа первого года обучения в основном ориентирована на формирование мощной теоретической базы, на основе которой будет проходить дальнейшее обучение. Дети изучают теоретические аспекты программирования (понятия переменной, оператора присваивания и других основополагающих понятий, без которых невозможно стать полноценным программистом), а также получают первый практический опыт программирования. Для повышения интереса детей к обучению в программу введено написание игровых программ, что гораздо ближе ребенку, чем прикладные программы, связанные с математикой. Переработка программы вызвана необходимостью увеличения количества часов на практику на 1 году обучения. Анализ результатов работы творческого объединения показал необходимость расширения практической части, так как освоение мощной теоретической базы требует более детального практического подкрепления.

На втором году обучения, на основе преемственности, необходимо закрепить знания, полученные на первом этапе. Практика – вот единственный критерий, по которому можно оценить уровень развития программиста. Именно поэтому почти полгода выделяется на самостоятельную работу над творческим курсовым проектом. И лишь после качественного освоения предыдущего материала можно перейти к освоению системного программирования и языка ассемблер, строения операционной системы MS DOS, архитектуры ЭВМ. Изучение столь сложных понятий не является самоцелью, оно лишь помогает понять те процессы, которые протекают в ЭВМ, позволяет писать программы не на «интуитивном» уровне, а на осознанном уровне, полностью отдавая себе отчет, почему все происходит так, а не иначе. Основная задача второго года обучения – через практику научить школьников самостоятельно искать ошибки в программах. Для достижения поставленных целей используется метод проектов.

На третьем году обучения дети получают возможность участвовать во всех городских конкурсах, олимпиадах и конференциях по программированию, и за счет полученных ранее знаний, написать серьезные программы прикладного назначения. Первое полугодие ребята пишут творческий курсовой проект и защищают его на конференции по программированию, традиционно проводимой в Центре в январе месяце. Для этого им необходимо создать прикладную или игровую программу высокого уровня сложности. Язык, на котором она будет реализована, выбирается самостоятельно. Обычно это ассемблер. Во втором полугодии кружковцы изучают объектно-ориентированное программирование (ООП), строение и функционирование сетей ЭВМ. **Объектно-ориентированный подход – это совершенно новое и важнейшее направление в современных технологиях программирования.** Почему же не начать работу с него? Дело в том, что прежде чем начать эту сложнейшую и наиболее важную тему, необходимо подготовить теоретическую и практическую базу в сознании учащегося, чтобы «посеянные семена» дали хорошие всходы. Чтобы получить из ученика творчески мыслящего программиста, а не ремесленника, который на неосознанном интуитивном уровне «вывает» программы. Основная задача третьего года – научить школьников самостоятельно добывать, обрабатывать и использовать информацию для достижения поставленных целей. Огромное внимание на третьем году уделяется самостоятельной исследовательской работе.

Обучающиеся, успешно освоившие программу обучения, сдают экзамен в соответствии с квалификационной характеристикой и получают свидетельство о прохождении курса «Системное программирование».

Цель работы объединения:

- *Развитие интеллектуальных и творческих способностей учащихся*

Для достижения этих целей необходимо решить несколько промежуточных задач:

Задачи обучения:

1. Изучить теоретические и практические основы программирования, архитектуру ЭВМ, различные технологии программирования;
2. Познакомить учащихся с этапами разработки творческих проектов по программированию.

Задачи развития:

1. Формировать творческое мышление и мировоззрение;
2. Вести предпрофессиональную подготовку учащихся, способствовать выбору будущей профессии.

Задачи воспитания:

1. Формировать доброжелательные отношения друг к другу;
2. Формировать у учащихся умение работать в команде, распределять роли в ней, планировать свою деятельность в рамках команды.

1.2. Учебно-тематический план

План работы в творческом объединении предусматривает обучение в течение трех лет.

Первый год:

- формирование у учащихся навыков, характеризующих грамотных пользователей ЭВМ;
- изучение основ программирования и основных этапов в написании программ;
- изучение структур данных и основных приемов программирования.

Второй год:

- предпрофессиональная подготовка;
- закрепление теоретических знаний и их практическая отработка при создании творческого курсового проекта;
- более глубокое изучение архитектуры ЭВМ;
- изучение языка символического кодирования ассемблер;
- изучение системного программирования.

Третий год:

- закрепление полученных ранее знаний в области системного программирования и их практическая отработка при создании творческого курсового проекта;
- изучение объектно-ориентированного подхода к программированию;
- участие в городских, областных и Российских конкурсах, олимпиадах и конференциях по программированию;
- изучения строения и функционирования локальных вычислительных сетей.

№	Тема	Количество часов		
		Всего	Теория	Практика
<u>I год обучения</u>				
I.	Алгоритмические языки. Основные алгоритмические приемы при решении задач.	6	3	3
II.	Основы языка программирования Pascal.	20	10	10
III.	Создание игровых программ.	20	10	10
IV.	Язык программирования Паскаль	20	10	10
V.	Структуры данных.	58	24	34
VI.	Решение практических задач на ЭВМ	10	5	5
VII.	Резервное время. Контроль ЗУН.	10	0	10
<u>Итого</u>		144	62	82
<u>II год обучения</u>				
VIII.	Язык символического кодирования assembler для процессора Intel 8086.	38	19	19
IX.	Периферийные контроллеры.	44	22	22
X.	Системное программирование.	52	26	26
XI.	Резервное время. Контроль ЗУН.	10	0	10

	Итого	144	67	77
	<u>III год обучения</u>			
XII.	Объектно-ориентированный подход к программированию на примере языка Паскаль.	30	15	15
XIII.	Написание интерфейсной оболочки на ООП Pascal.	60	30	30
XIV.	Строение и функционирование сетей ЭВМ.	40	20	20
XV.	Резервное время. Контроль ЗУН.	14	0	14
	<u>Итого</u>	144	65	79

1.3. Содержание программы

Первый год обучения

I. Алгоритмические языки. Основные алгоритмические приемы при решении задач (6 часов)

Смысл и назначение данного раздела программы обучения - научить детей правильно определять и строить основные структуры программы: следование, ветвление, цикл. Важность этого раздела невозможно переоценить. На начальном этапе детям очень сложно правильно определять тип цикла в конкретной задаче, последовательность, порядок шагов при ее решении, условие, при котором необходимо выполнить то или иное действие.

1) Понятие алгоритма, его свойства, виды программ, трансляторы. Понятие переменной.

В результате обучения учащиеся должны:

знать: понятие переменной, ее смысл, операции, которые возможно выполнить над ней.

2) Оператор присваивания.

знать: порядок выполнения действий при присваивании переменной значений, смысл переменной слева и справа от оператора присваивания.

3) Оператор ветвления.

знать: порядок действия оператора ветвления, полную и сокращенную форму оператора ветвления.

уметь: решать простые задачи на использование оператора ветвления, определять результат работы алгоритма.

форма контроля знаний: самостоятельные работы.

4) Циклы: с известным числом повторений, с постусловием, с предусловием.

знать: порядок работы каждого из циклов, максимальное и минимально количество выполнений тела цикла, условие продолжения и завершения работы циклов.

уметь: решать простые задачи с использованием трех видов циклов, определять результат работы алгоритма.

форма контроля знаний: самостоятельные работы.

II. Основы языка программирования Pascal (20 часов).

В данном разделе даются первоначальные сведения о языке программирования Паскаль, при постепенном усложнении задач, решаемых на блок-схемах, происходит одновре-

менный перевод их на Паскаль. Одной из важных задач данного раздела - показать сходность решенных задач на языке блок-схем и Паскаля. Ребенок должен уметь переводить программу с блок-схемы на Паскаль и обратно.

1) Понятие типа данных. Основные типы данных (целый, вещественный, литерный).

знать: основные типы в Паскале, их значения и операции, выполнимые над ними. Представление в памяти переменных основных типов.

уметь: определить тип переменной по ее значениям и операциям над ней.

форма контроля знаний: самостоятельные работы.

2) Оператор ветвления.

знать: полную и сокращенную форму оператора ветвления, в каких случаях необходимо ставить логические скобки *begin - end*, правила постановки знака “;”.

уметь: переводить оператор ветвления с блок-схемы на Паскаль.

форма контроля знаний: самостоятельные работы.

3) Интегрированная среда Turbo Pascal.

знать: основные возможности среды и горячие клавиши основных функций.

4) Операторы цикла.

знать: синтаксис трех видов циклов, условия их применения, их достоинства и недостатки.

уметь: определить тип цикла, который должен использоваться при решении данной задачи, правильно переводить структуру цикл с блок-схемы на Паскаль.

форма контроля знаний: самостоятельные работы.

5) Оператор безусловного перехода GOTO: его вред для программирования.

знать: последствия применения оператора безусловного перехода GOTO.

форма контроля знаний: самостоятельные работы.

6) Графический режим Паскаля.

знать: режимы работы дисплея, порядок инициализации графического режима, название и назначение основных графических функций.

уметь: строить на экране графическое изображение.

форма контроля знаний: практические работы:

- программа «Домик»;
- программа «Дым из трубы»;
- программа «Паровоз»;
- программа «звездное небо» (генератор случайных чисел);
- программа «Гусеница», движение гусеницы;
- параллельное движение, игра «Солнечная система»;
- Программа «Часы».

III. Создание игровых программ. (20 часов)

Данный раздел предназначен для закрепления полученных знаний, он знакомит с основными принципами мультипликации, с организацией параллельного движения картинок на экране. Выбор в качестве базовых программ - игровых, обусловлен следующими факторами:

- интерес ребенка к процессу создания игры;
- сложность организации игровых программ;
- большая перспектива развития данного направления программирования.

1) Параллельное движение объектов по экрану: игра "Охота на кролика".

знать: принципы организации параллельного движения нескольких картинок по экрану, функции опроса клавиатуры без ожидания, методы организации случайных событий.

уметь: заставить картинку двигаться по экрану, разработать структуру данных для организации движения двух картинок.

форма контроля знаний: творческая работа - игра "Охота на кролика". По экрану одновременно двигаются два персонажа: "кролик", управляемый компьютером и "охотник", управляемый с клавиатуры. Задача "охотника" - убить "кролика". Кролик через случайное время может ставить бомбы, которые, взрываясь, убивают "охотника", если тот находится в радиусе поражения.

2) Работа с массивами.

знать: строения массива, способы его описания, методы работы с ним. Признаки необходимости использования массива.

уметь: использовать массив при решении задач.

форма контроля знаний: самостоятельные работы.

3) Линейный список и операции над ним (создание, вывод, добавление, удаление элементов, поиск по ключу, сортировка элементов. Оценка скоростных характеристик поиска информации, поиск с барьером, бинарный поиск).

знать: возможные операции и методы их реализации в линейном списке, способы организации линейного списка, методы поиска в линейном списке, их достоинства и недостатки, скоростные характеристики различных алгоритмов поиска, методы сортировок линейных списков, скорость и эффективность сортировок.

уметь: организовать, разработать структуру данных для хранения линейного списка, написать алгоритм, фрагмент программы для выполнения различных операций над линейным списком

форма контроля знаний: практическая работа. Программа "База данных". В БД хранится информация о некоторых объектах (учениках, машинах, продукции). БД позволяет ввести информацию об объектах, добавить, удалить, отсортировать информацию, найти информацию по ключу.

4) Директивы компилятора

5) Параллельное движение случайного количества объектов по экрану: игра "Тренажер клавиатуры".

знать: методы организации и основные операции над линейным списком, механизм организации движения изменяющегося количества объектов по экрану.

уметь: заставить двигаться постоянно меняющееся количество картинок, создавать и удалять объект с экрана, организовывать взаимодействие между объектами (столкновение, преследование и т. п.).

форма контроля знаний: творческая работа - игра "Тренажер клавиатуры". С верхней части экрана падают буквы, их количество постоянно меняется (то 0 до 7). Максимальное количество букв, которые могут одновременно находиться на экране, зависит от уровня игры. Если пользователь нажимает соответствующую букву на клавиатуре, то она "сбивается", то есть удаляется с экрана и через некоторое время появляется новая буква.

5) Строковый тип.

знать: строения строки, способы ее описания, основные приемы работы с ней, функции работы со строками.

уметь: организовывать процесс ввода и вывода строк, уметь решать задачи по теме «Строки».

6) Двумерные массивы (матрицы)

знать: строения матрицы, способы ее описания, основные приемы работы с ней.

уметь: организовывать процесс ввода и вывода данных из матрицы, уметь решать задачи по теме «Матрицы».

форма контроля знаний: самостоятельные работы.

7) Движение объекта в среде с препятствиями: игра «Красная шапочка».

Знать: основные принципы организации движения объекта в среде с препятствиями, понятие массива-образа экрана, относительных и экранных координат объекта, взаимное

преобразование координат, метод генерации случайного игрового поля, механизм определения препятствия переддвигающимся объектом.

Уметь: организовать структуру данных для моделирования движения объекта в среде с препятствиями, генерировать игровое поле, разрабатывать структуру данных длядвигающегося объекта.

Форма контроля знаний: творческая работа – игра “Красная шапочка”. По игровому полю, на котором разбросаны препятствия и предметы сбора (монеты, цветки и т. П.) двигается главный персонаж – “Красная шапочка”. Если впереди нее препятствие, то она должна остановиться, если предмет сбора, то увеличивается количество очков, а данный предмет удаляется с экрана. За главным героем охотится противник – “Серый волк”, его задача съесть “Красную шапочку”.

8) Отладка программ в интегрированной среде Turbo Pascal 7.0

знать: основные команды среды, горячие клавиши.

уметь: выполнять пошаговое исполнение программы, наблюдать за состоянием переменных.

IV. Язык программирования Паскаль (20 часов).

Данный раздел продолжает знакомство обучаемых с языком программирования Паскаль. Задачи раздела:

- Познакомить обучающихся с новыми структурированными данными;
- За счет расширенной практики освоить приемы программирования;
- Познакомить учащихся с отладчиком и способами поиска логических ошибок в программах.

1) Подпрограммы в языке Паскаль

знать: понятие подпрограммы, требования к подпрограммам, виды подпрограмм, отличие процедур и функций, знать способы передачи параметров подпрограмма, понятие локальных и глобальных подпрограмм.

уметь: создавать и использовать подпрограммы.

2) Файлы в языке Паскаль

знать: понятие файла, типы файлов, строение файла, основные методы работы с ним.

уметь: создавать, читать и дописывать информацию в файл.

3) Множества и операции над ним.

V. Структуры данных. (58 часов).

Данный раздел позволяет познакомить детей с основными структурами данных и основными алгоритмическими приемами при решении поставленных задач. Для его успешного восприятия необходим уже высокий уровень знания и понимания языка Паскаль, а также некоторый опыт практического программирования на ЭВМ. Знания, полученные на этом этапе, будут использованы на втором и третьем году обучения при подготовке к конкурсам по программированию.

1) Стек и операции над ним, очередь и операции над ней.

знать: понятие и назначение стека, возможные операции над ним, области применения структуры данных - стек. Механизм работы очереди, операции над ней, использование очереди в программах.

уметь: разработать структуру данных и организовать стек и очередь в программе.

форма контроля знаний: практическая работа. Анализ скобочной структуры с использованием стека.

2) Динамический линейный список и операции над ним.

знать: понятие указателя и ссылочного типа, общие принципы работы с динамическими переменными, порядок описания и создания динамического списка, основные операции над динамическим линейным списком (создание, удаление, вывод, поиск, добавление звена),

преимущества и недостатки динамических списков по сравнению со статическими. Виды линейных динамических списков, их достоинства и недостатки.

уметь: разработать структуру данных для представления динамического списка, написать фрагменты программы для реализации основных операций над динамическим линейным списком.

форма контроля знаний: практическая работа. В линейном динамическом списке хранится символьная информация. Программа позволяет ввести предложение, добавить, удалить, букву, найти информацию по ключу.

3) Разреженные матрицы

знать: понятие разреженной матрицы, способы ее более компактного представления в памяти ЭВМ.

уметь: организовать в программе представление разреженной матрицы и работу с ней.

форма контроля знаний: практическая работа.

4) Преобразование арифметических выражений в обратную польскую запись

знать: алгоритм перевода обычного арифметического выражения в обратную польскую запись и выполнения расчетов в ней.

форма контроля знаний: творческая работа. Написать программу выполнения расчетов арифметических выражений, вводимых с клавиатуры

5) Хеш-функции.

знать: понятие и назначение Хеш-функции, области ее применения.

уметь: создать Хеш-функцию для конкретной задачи.

форма контроля знаний: практическая работа. Создать Хеш-функцию для ускорения поиска в некотором массиве.

6) Механизм вызова подпрограмм

знать: порядок, механизм вызова подпрограммы.

7) Рекурсивные подпрограммы

знать: понятие, строение и возможность применения рекурсивных подпрограмм.

уметь: создавать рекурсивные подпрограммы.

форма контроля знаний: практическая работа.

8) Двоичная система счисления.

знать: понятие систем счисления, их виды и отличия, правила перевода числа из одной системы счисления в другую. Знать единицы измерения объемов информации.

уметь: переводить числа из одной системы счисления в другую, преобразовывать единицы объема информации, определять ее объем.

форма контроля знаний: контрольная работа.

9) Бинарные деревья, операции над ними.

знать: понятие, назначение и строение структуры "дерево", виды деревьев и их отличительные особенности, основные операции над деревьями (создание, поиск, добавления узла), основные части деревьев, алгоритм сжатия информации по методу Хаффмана.

уметь: создавать дерево поиска.

форма контроля знаний: Программа сжатия информации по алгоритму «Хаффмана».

VI. Решение практических задач на ЭВМ. (10 часов).

Данный раздел предназначен для закрепления навыков программирования, полученных на предыдущих этапах. Большое внимание в данном разделе уделяется самостоятельной работе обучающегося за ПК.

- нахождение площади фигур методом «Монте-Карло»;
- построение графиков функции в декартовых координатах;
- построение графиков функции в полярных координатах;
- нахождение корней уравнения методом половинного деления;

- нахождение значения интеграла методом трапеции;
- поиск пути в лабиринте;
- разбор арифметических выражений с использованием бинарных деревьев;
- конечные автоматы.
- Модуль VESA, использование 256-цветного режима работы дисплея.

форма контроля знаний: *практические работы.*

VII. Резервное время. Контроль ЗУН. (10 часов).

Участие в мероприятиях различного уровня. Начальный контроль ЗУНов проводится на предварительном тестировании, промежуточный, итоговый – в рамках обозначенных тем.

Второй год обучения

Второй год обучения структурно делится на две части: первые 96 часов отводятся для самостоятельной работы для разработки творческого проекта к конференции и конкурсу творческих работ по программированию в ЦДТ, который проходит ежегодно в конце декабря. Задачей этого этапа является освоение и практическое закрепление знаний, полученных на предыдущем году обучения. Для этого используется творческая работа учащегося на свободную тему. Она включает все этапы создания программного обеспечения: разработка сценария, разработка структуры данных и математической модели, написание алгоритмов, перевод их на Паскаль, отладка программы, создание реферата, защита программы на конференции по программированию.

Следующие 110 часов отводятся под очень глубокое изучение архитектуры IBM компьютеров, строения и основных управляющих структур операционной системы MS DOS. В качестве метода контроля полученных знаний используются лабораторные работы, проводимые по всем основным темам. Столь подробное изучение архитектуры и низкоуровневого программирования не является самоцелью, оно лишь призвано сформировать у учащихся творческое мировоззрение, понимание процессов, протекающих при работе ПК, чтобы в дальнейшей работе они могли использовать полученные знания для осознанного, а не интуитивного программирования.

VIII. Язык символического кодирования assembler для процессора Intel 8086. (38 часов)

Целью данного раздела является глубокое изучение языка низкого уровня - ассемблер. Знание этого языка потребуется для работы с аппаратной частью ЭВМ и со структурами операционной системы. Полученные знания позволяют взглянуть на процесс создания программ как бы изнутри, с точки зрения процессов, протекающих внутри ПК и операционных систем, позволяют понять, как представляются в памяти различные переменные, массивы, что такое подпрограмма, как она организована с точки зрения аппаратуры. Здесь изучается организация оперативной памяти, понятие сегмента, адресация, назначение регистров.

1. **Архитектура фон Неймана, архитектура процессора Intel 8086**
2. **Адресация, понятие сегмента**

знать: *способы адресации, понятие сегмента.*

уметь: *вычислять физический адрес переменной по имеющемуся логическому.*

3. **Операторы пересылки данных.**

знать: *названия и назначения регистров процессора, синтаксис команды MOV, ее операнды.*

уметь: *пересылать данные в регистры и память, выделять необходимое количество памяти, вводить переменные.*

4. **Операторы условного и безусловного перехода.**

знать: *название и назначение флагов процессора, синтаксис операторов условного и без-*

условного перехода, их аппаратную реализацию.

уметь: *выполнять сравнение величин и выполнять переходы на различные метки.*

5. Арифметические операции.

знать: *синтаксис арифметических операций и их операнды, представление чисел в двоичной, шестнадцатеричной системах.*

уметь: *выполнять арифметические операции над многобайтными числами.*

6. Строковые операции.

знать: *синтаксис и используемые регистры в цепочечных командах MOV, STOS, LODS, SCAS, знать порядок работы префикса REP, назначение флага DF.*

уметь: *пересылать данные между областями памяти.*

7. Дополнительные операторы.

форма контроля знаний: *практическая работа, лабораторные работы.*

8. Массив, работа с массивом

знать: *создание массивов, основные операции над ним, представление массивов в ОЗУ.*

уметь: *вводить и выводить данные из массива*

9. Пошаговое исполнение программ. Отладчики.

IX. Периферийные контроллеры. (44 часа)

Данный раздел посвящен изучению аппаратного строения IBM PC, здесь изучаются основные периферийные контроллеры, управляющие различными внешними устройствами. Полученные знания позволяют напрямую управлять аппаратурой, минуя операционную систему, что позволяет создавать более эффективные программы, восстанавливать ЭВМ, в случае необходимости. Изучение аппаратного строения ЭВМ приводит к более высокому уровню понимания проблем создания программ.

1) таймер;

знать: *назначение и аппаратное строение таймера, регистры общения с ним, механизм вывода звука на динамик и использование при этом таймера.*

уметь: *выводить звук в фоновом режиме работы программы, изменять режимы работы таймера.*

форма контроля знаний: *лабораторные работы. Сирена - генерация звука с использованием таймера. Часы - вывод системного времени в реальном режиме. Подсчет времени работы программы.*

2) контроллер прерываний;

знать: *назначение и аппаратное строение контроллера прерываний, регистры общения с ним, механизм организации аппаратных прерываний.*

форма контроля знаний: *контрольная работа.*

3) контроллер клавиатуры;

знать: *назначение и аппаратное строение контроллера клавиатуры, регистры общения с ним, обработка нажатия и отпускания клавиши, механизм активизации 9^{го} прерывания.*

уметь: *перехватывать прерывание от клавиатуры, анализировать нажатие и отпускание клавиши.*

форма контроля знаний: *лабораторная работа. Программа "Орган" - звуковая клавиатура. При нажатии на клавишу идет звуковой сигнал (отличный у всех клавиш), при отпускании клавиши - звук прекращается.*

4) видеоадаптер;

знать: *назначение и аппаратное строение видеоадаптера, регистры общения с ним, режимы работы дисплея, отображение видеопамати в различных режимах работы. Адаптеры CGA, EGA, VGA и SVGA, VESA стандарт.*

уметь: *переводить дисплей в различные режимы работы, выводить текстовую информацию в текстовом режиме, непосредственной записью в видеобуфер, выводить информацию в графическом режиме.*

форма контроля знаний: *лабораторная работа.*

Текстовый режим: Прямой вывод в видеобуфер, Светомузыка, бегущая строка, программа «Вертолет».

графический режим: Вывод информации в графическом режиме.

5) контроллер диска.

***знать:** логическое и физическое строение диска, назначение и расположение Boot Record, FAT, Root Directory. Понятие кластера, механизм записи информации в файл*

***уметь:** находить на диске различные части системной области.*

***форма контроля знаний:** лабораторная работа. Изучение структуры диска при помощи редактора диска.*

X. Системное программирование. (52 часа)

В данном разделе глубоко изучается строение и функционирование операционных систем на примере ОС MS DOS. Изучаются основные структуры ОС, их взаимодействие, связь программы и ОС, механизмы запусков программ, строение EXE и COM файлов, строение и функционирование резидентных программ и драйверов внешних устройств.

Полученные знания помогут работать не только в операционной системе MS DOS, но и в любой другой, так как для перехода к другой ОС достаточно будет прочитать документацию на нее и определить некоторые их отличия.

1) Модульное программирование.

Понятие сегмента, параметры объединения сегментов, выравнивание сегментов, их расположение в оперативной памяти, общие и личные переменные, внешние переменные, связь языков Паскаль и ассемблер.

***знать:** понятие модуля, способы оформления модуля, достоинства и недостатки модульного программирования, способы описание внешних переменных, экспорт и импорт переменных.*

***уметь:** формировать модули, уметь объединять их, связывать модули на Паскале и ассемблере.*

***форма контроля знаний:** лабораторная работа. Связь Паскаля и ассемблера, вызов подпрограмм, передача данных подпрограмма (через память, стек, PОНы). Различные способы объединения сегментов, параметры выравнивания сегментов.*

2) Командная строка. PSP-процесса.

***знать:** назначение и месторасположения PSP-процесса, его строение.*

***уметь:** определять параметры программы в PSP, уметь анализировать командную строку.*

***форма контроля знаний:** лабораторная работа. Вывод командной строки на экран.*

3) Окружение процесса.

***знать:** назначение и месторасположения окружения процесса.*

***уметь:** анализировать и изменять окружение процесса.*

***форма контроля знаний:** лабораторная работа. Вывод окружения процесса на экран.*

4) Организация оперативной памяти. МСВ-блоки.

***знать:** организацию оперативной памяти, структуру МСВ-блока, операции с блоками памяти: освобождение, выделение, изменения размеров.*

***уметь:** перемещаться по цепочке блоков, определять их тип и принадлежность, выполнять основные операции с блоками памяти.*

***форма контроля знаний:** лабораторная работа. Программа, которая просматривает блоки памяти и выводит о них следующую информацию: адрес, размер, тип блока, кому принадлежит.*

5) Работа с файлами на ассемблере.

***знать:** порядок создания, записи, чтения, удаления и переименования файла, два способа работы с файлами: метод FCB и метод файловых манипуляторов.*

***уметь:** выполнять операции при работе с файлами.*

форма контроля знаний: лабораторная работа. Создание, чтение информации из файла.

6) Построение компилятора, компоновщика, формат объектного файла.

знать: строение, назначение, механизм работы компилятора и компоновщика.

Иметь представление о структуре объектного файла.

7) Форматы EXE и COM файлов.

знать: отличия в строении EXE и COM файлов, структуру заголовка EXE файла.

уметь: программно отличать EXE и COM файлы, уметь создавать их на ассемблере.

форма контроля знаний: лабораторная работа.

8) Вызов одной программы из другой.

знать: алгоритм вызова одной программы из другой, способы вызова программ.

уметь: программно загружать с диска программу, уметь ее запускать.

форма контроля знаний: лабораторная работа. Вызов одной программы из другой. Имя дочернего процесса задается в командной строке родителя.

9) Загрузка оверлеев.

знать: понятие оверлея, его отличие от программы, алгоритм создания и загрузки оверлея.

уметь: создавать оверлейные модули и подгружать их в оперативную память по мере необходимости.

форма контроля знаний: лабораторная работа. Написать программу, которая загружает два типа оверлеев: EXE и COM формата.

10) Резидентные программы.

Строение резидентной программы, перехват векторов прерываний, проблема повторного запуска, требования к обработчикам прерываний, уровни сложности TSR-программ, нереентерабельность DOS, тайна int 28h.

знать: методы решения проблемы нереентерабельности MS DOS, схему создания TSR-программы второго уровня сложности, схему взаимодействия обработчиков прерываний, методы обнаружения резидента в памяти и выгрузки из ОЗУ.

уметь: создавать резидентные программы второго уровня сложности.

форма контроля знаний: лабораторная работа. Написание TSR-программы второго уровня сложности: копировщик экрана, защитник файлов и т. д.

11) Драйверы внешних устройств.

Понятие драйвера, их виды, структура драйвера, заголовков драйвера, процедуры стратегии и обработки прерываний, функция инициализации, запросы к драйверам.

знать: строение и основные части драйвера, основные функции и запросы к драйверу, отличия символьных и блочных драйверов.

уметь: создавать драйверы внешних устройств.

форма контроля знаний: лабораторная работа. Написание драйвера клавиатуры, который преобразует строчные буквы в заглавные.

12) Творческая работа. Самостоятельная разработка проекта TSR-программы или драйвера ВУ.

XI. Резервное время. Контроль ЗУН. (10 часов).

Участие в мероприятиях различного уровня. Начальный промежуточный, итоговый контроль ЗУНов проводится в рамках обозначенных тем.

Третий год обучения

Третий год обучения структурно делится на две части: первые 104 часа отводятся для выполнения самостоятельной творческой работы в области системного программирования. Задачей этого этапа является освоение и практическое закрепление ранее полученных знаний. Творческая работа включает все этапы создания программного обеспечения: разработка сценария, разработка структуры данных и математической модели, написание алгоритмов, перевод их на язык программирования, отладка программы, создание реферата, защита программы на конкурсе. В качестве темы работы обычно выбирается сложная системная программа: резидентная программа, драйвер, работа с видеоадаптером, мышкой, принтером. Выбор темы осуществляет сам старшеклассник, причем в процессе создания программы, он самостоятельно более глубоко изучает выбранную тему и связанную с ней аппаратную часть ЭВМ, и программное обеспечение. Это позволяет не только закрепить полученные ранее знания, но и существенно их расширить и углубить.

Следующие 112 часов отводятся для изучения двух наиболее перспективных современных направлений развития программирования: основы работы компьютерных сетей и объектно-ориентированного программирования. На этом этапе изучаются основные принципы передачи информации между ЭВМ, методы организации локальных вычислительных сетей, модемной связи, некоторые стандарты и протоколы обмена в ЛВС, методы защиты информации от ошибок и сбоев при ее передаче. При изучении объектно-ориентированного программирования основное внимание уделяется практической отработке полученных знаний. Это связано с тем, что изучается новый подход к уже известному языку Паскаль. Здесь изучается понятие объекта, его возможности: наследование, полиморфизм, инкапсуляция, взаимодействие объектов, организация событий и их обработка.

ХII. Объектно-ориентированный подход к программированию на примере языка Паскаль. [30 часов]

- 1) **Понятие объекта.** Задачи, решаемые с помощью ООП, разбиение задачи на подзадачи, “исполнители” и сообщения.
 - 2) **Основные свойства объектов. Инкапсуляция, наследование, полиморфизм.**
 - 3) **СRС- карточки.** Методы создания сложного программного обеспечения при участии нескольких человек, взаимодействие между программистами, разбиение сложной задачи на подзадачи, распределение подзадач между программистами.
 - 4) **Методы и классы.** Понятие методов, инкапсуляция методов, скрытие информации внутри объекта, обработка событий.
 - 5) **Сообщения.** Обмен сообщениями между объектами, передача и прием сообщений.
 - 6) **Инициализация объектов.** Понятие конструктора и деструктора, явный вызов конструктора, создание динамических объектов и их инициализация.
 - 7) **Наследование, повторное использование кода.** Создание дочерних классов, наследование полей и методов, упрощение программ за счет повторного использования кода.
 - 8) **Подклассы, замещение и уточнение методов.** Изменение функций некоторых методов, вызов методов предка, полное замещение методов предка.
 - 9) **Виртуальные методы.** Понятие виртуального метода, его описание, механизм вызовов виртуальных методов.
- форма контроля знаний: экзамен.**

ХIII. Написание интерфейсной оболочки на ООП Pascal. [60 часов]

Данный раздел предназначен для отработки на практике полученных теоретических знаний. Старшеклассники создают свою программу на ООП Паскале, которую в дальнейшем они смогут использовать для создания интерфейса в своих программах.

форма контроля знаний: защита творческой работы на экзамене.

ХIV. Строение и функционирование сетей ЭВМ. [40 часов]

- 1) **Этапы развития сетей.**
 - 2) **Топология сетей.** Виды топологии сетей, их достоинства и недостатки, способы подключения к сети рабочих станций.
 - 3) **Программное обеспечение сетей.** Уровни программного обеспечения, взаимодействие уровней, модульный характер построения уровней.
 - 4) **Передача информации в сетях.** Способы передачи, физические среды, используемые для передачи информации, разбиение информации на блоки, фрагменты, пакеты, заголовки блоков, концевики блоков.
 - 5) **Кодирование и защита от ошибок, исправление ошибок.** Методы кодирования, сжатия информации, контроль при передаче информации, обнаружение ошибок, методы восстановления информации после ошибок, повторная передача пакетов, подсчет контрольных сумм.
 - 6) **Методы коммутации.** Коммутация станций, виды коммутаций, проблема выбора маршрута для передачи сообщения, подтверждение приема сообщения.
 - 7) **Модемы.** Модуляторы и демодуляторы, аналоговые и дискретные сигналы, связь по телефонной линии, достоинства и недостатки модемов.
 - 8) **Локальные вычислительные сети.** Организация сетей, методы передачи информации в сетях, сети с маркерным доступом, кольцевые сети с маркерным доступом, сети со случайным доступом.
 - 9) **Моноканал.** Понятие моноканала, его организация, передача информации в моноканале.
 - 10) **Основные стандарты обмена информацией в ЛВС (802.3, 802.4, 802.5).** Основные принципы функционирования сети, способы передачи информации, логика взаимодействия станций, защита от сбоев, разбиение информации на пакеты, методы коммутации в сети.
 - 11) **IPX-протокол.**
- форма контроля знаний:** творческая работа “Передача информации по ЛВС с использованием протокола IPX”.

XV. Резервное время. Контроль ЗУН. (14 часов).

Участие в мероприятиях различного уровня. Начальный промежуточный, итоговый контроль ЗУНов проводится в рамках обозначенных тем.

1.4. Планируемые результаты

В результате обучения в объединении программирования по данной программе должны быть получены следующие результаты:

учащиеся должны знать: строение и основные части ЭВМ, операторы алгоритмического языка блок-схема, их назначение и порядок работы, синтаксис языка программирования Паскаль и правила работы в интегрированной среде Turbo Pascal 7.0, теоретические основы организации параллельного движения нескольких объектов, организации движения случайного количества объектов, движение объектов в среде с препятствиями, методы хранения информации и методы повышения эффективности работы с ней, основные структуры данных, синтаксис языка низкого уровня ассемблер, методы объединения программ, написанных на разных языках, организация взаимодействия между ними, понятие модуля, порядок оформления модулей, архитектуру ЭВМ, периферийные контроллеры: таймер, клавиатура, диск, видеоадаптер, СОМ-порт, методы взаимодействия и управления контроллерами внешних устройств, порядок создания и внутреннее строение резидентных программ и драйверов внешних устройств, назначение, функционирование локальных вычислительных сетей, методы, способы передачи информации в сетях, кодирование и защита информации при ее передаче в ЛВС, основные стандарты передачи информации в сетях 802.3, 802.4,

802.5, объектно-ориентированный подход к программированию, понятие объекта, его свойства, способы активизации, распространения компьютерных вирусов, их виды, методы защиты от вирусов, их обезвреживание, ликвидация последствий деятельности вирусов.

Должны уметь: выполнять действия, направленные на поддержание ЭВМ в работоспособном состоянии, обеспечивать распределение ресурсов ЭВМ между пользователями, распределять права, восстанавливать работоспособность ЭВМ после повреждения программного обеспечения, разбивать винчестер на логические диски, форматировать логические диски винчестера, устанавливать новое программное обеспечение, системы защиты информации и распределения ресурсов, настраивать работу стандартного программного обеспечения, создавать программы на языках Паскаль и ассемблер, использовать прямые обращения к внешним устройствам для повышения быстродействия и эффективности, создаваемого программного обеспечения, использовать приемы объектно-ориентированного программирования при создании программ высокого уровня сложности.

2. Комплекс организационно-педагогических условий

2.1. Календарный учебный график

1 год обучения

Количество часов в неделю	Количество часов в месяц	Количество часов в год
4	16	144

2 год обучения

Количество часов в неделю	Количество часов в месяц	Количество часов в год
4	16	144

3 год обучения

Количество часов в неделю	Количество часов в месяц	Количество часов в год
4	16	144

2.2. Методическое обеспечение

Программа первого года обучения рассчитана на нулевой уровень начальных знаний учащихся. Естественно, что в первые два - три месяца изучения совершенно нового и не похожего на другие науки предмета, у учащегося возникают проблемы психологического характера. Это связано с особенностями функционирования человеческого мозга и отличиями его мышления от деятельности ЭВМ. Для решения возникающих проблем технологию обучения можно разделить на следующие части:

1) Первые два - три месяца проводится совместное решение задач одного типа. Урок состоит из двух частей. В начале преподаватель решает задачу на доске, анализирует ход ее решения, указывает на возможные ошибки. Затем ребята самостоятельно пытаются решить аналогичную задачу. После решения проводится совместный разбор ее решения у доски. Подобный прием позволяет научить мозг учащихся основным подходам к решению задач.

2) Следующим подходом к обучению является обозначение проблемы перед ребенком. Перед учащимися ставится конкретная задача, совместно с ним анализируется ее условие, не вдаваясь глубоко в детали, то есть, мы совместно решаем, что необходимо сделать для достижения поставленной цели. А затем ученик самостоятельно решает, как конкретно реализовать поставленную задачу. Такой подход позволяет научить старшеклассника самостоятельно мыслить, самому правильно строить структуру программы. Только человек успешно прошедший этот этап обучения сможет в дальнейшем хорошо справляться с поставленными перед ним задачами.

3) Третий этап в технологии обучения является самостоятельная работа учащегося. Перед ним ставится конкретная задача. Выбор ее темы осуществляется совместно, чтобы заинтересовать ребенка ее решением. Кружковец самостоятельно решает, что необходимо сделать для достижения поставленной цели, и какие для этого необходимо применить способы. Такой подход вовсе не означает, что дети предоставлены сами себе. Со стороны преподавателя должен осуществляться постоянный контроль за тем, как ученик пытается решить поставленные перед ним задачи, какие пути он при этом выбирает. Необходимо помочь ему найти правильный путь, но при этом нельзя "толкать" его в нужном направлении. Пусть он пойдет в обход и убедится на своем опыте, что этот путь хуже. Зато впоследствии, уже опираясь на свой опыт, он будет точно знать, что и как лучше сделать.

Для третьей формы работы лучше всего подходит самостоятельное написание игровых программ. Во-первых, ребенок учится жизни в процессе игры, она ему гораздо ближе. А во-вторых, другие сложные задачи (базы данных, моделирующие системы и тому подобное) ему просто не интересны.

Формы и методы обучения

При выборе форм и методов обучения следует решить, что вы хотите достигнуть: научить решать задачи, научить писать программы, научить думать или что-то еще. «Что такое интеллект?», «Как управлять или влиять на его формирование?». По мнению М.А. Холодной, ответ следует искать в другой плоскости – «что представляет собой интеллект как носитель своих свойств?» Носителем свойств интеллекта является индивидуальный ментальный (умственный) опыт. Здесь гораздо важнее не что думает данный человек, а как он это делает, как он осмысливает происходящее.

В когнитивном направлении психологии личности были обнаружены, зафиксированы и описаны определенные ментальные образования, которые контролируют и регулируют способ восприятия, понимания и интерпретации человеком происходящих событий. Эти структуры получили названия когнитивные схемы или понятийные системы. От того, как устроены эти структуры, зависят конкретные проявления интеллектуальной активности и личные свойства социального поведения человека. Ментальные структуры – это система

психических образований, которые в условиях познавательного контакта с действительностью обеспечивают возможность поступления информации о происходящих событиях и ее преобразование, управляют процессами переработки информации. Ментальные структуры составляют основу индивидуального ментального опыта, который накапливается, видоизменяется в ходе взаимодействия субъекта с окружающим миром. Ментальные структуры отвечают за актуализацию субъектного пространства отражения, в рамках которого и строится конкретный образ конкретной ситуации. Ментальные структуры, находясь в «свернутом» состоянии, при столкновении с любым внешним воздействием способны «развернуться» и организовать особое ментальное пространство. Ментальное пространство – это особая динамическая форма состояния ментального опыта, которая оперативно актуализуется в условиях осуществления человеком тех или иных интеллектуальных актов. Именно ментальное пространство определяет такие когнитивные стили как импульсивность – рефлексивность, узость – широта видимого поля, жесткий – гибкий познавательный контроль и т.д. Ментальная репрезентация – это актуальный мыслительный образ того или иного конкретного события, то есть субъективная форма «видения» происходящего. По мнению Кликса, неудачи в решении задач связаны с построением ошибочной или слишком громоздкой репрезентации проблемной ситуации. В итоге не удастся выделить ее существенные признаки. Как строится индивидуальная «ментальная картинка» происходящего? В когнитивной психологии часто встречается мнение, согласно которой репрезентация рассматривается как средство хранения и организации знаний в виде следов памяти, списка признаков, вербальных и визуальных схем.

Итак, ментальный опыт – это система наличных психических образований и инициируемых ими психических состояний, лежащих в основе познавательного отношения человека к миру. Ментальный опыт представлен в трех формах: ментальные структуры, ментальное пространство, ментальные репрезентации.

Таким образом, **интеллект** по своему онтологическому статусу – *это особая форма организации индивидуального ментального опыта в виде наличных ментальных структур, порождаемого ими ментального пространства отражения и строящихся в рамках этого пространства ментальных репрезентаций происходящего*. Следовательно, роль ментального опыта является критичной по отношению к процессам, происходящим во время интеллектуальной деятельности. Поэтому в условиях низкой сформированности ментальных структур любое действие окажется «погребенным в безмолвии индивидуального опыта» (Д. Брунер).

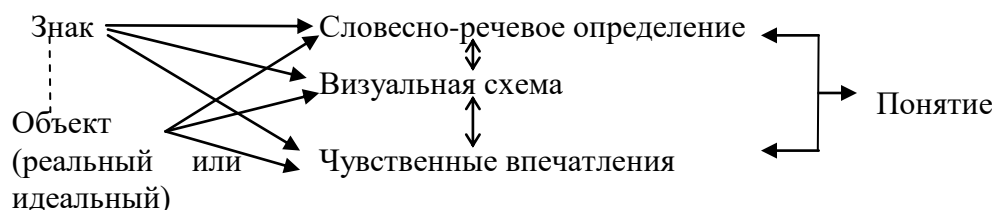


Л.С. Выготский, изучая закономерности умственного развития ребенка, пришел к заключению, что именно образование понятий является ключом к пониманию процессов психологического развития и распада. Образование понятий – длительный процесс. По Выготскому, собственно понятия появляются к 11-12 годам. Благодаря понятиям подросток начинает понимать связи, отношения взаимосвязи, скрытые за поверхностью видимых явлений. Кроме того, понятия – это средства упорядочивания воспринимаемого мира с помощью «сетки» категорийных и логических отношений. При помощи понятий происходит «расши-

рение среды сознания» подростка. Происходит перестройка элементарных познавательных функций на основе их синтеза с функцией образования понятий: восприятие фактически превращается в наглядное мышление, запоминание начинает опираться на смысловые связи, внимание приобретает произвольный характер. Благодаря формированию понятийного мышления содержание мышления становится внутренним убеждением подростка, его интересом, желанием и намерением.

Таким образом, мышление в понятиях обеспечивает возможность нового типа понимания объективного мира, возможность понимания других людей и возможность понимать самих себя.

Схематически образование понятий (по Выготскому) можно представить так:



Применительно к учебникам такая схема означает специальную организацию текста, связанную с формированием у учащихся способности к обратимому переводу информации с языка образов на язык визуальных схем. Чувственно-сенсорные составляющие понятийных структур могут быть сформированы за счет опоры в учебном тексте на предметно-жизненный опыт детей (актуализация их опыта из практической сферы, проведение аналогий, поиск примеров из реальной жизни).

В процессе формирования понятия можно выделить следующие фазы:

1. **мотивировка** – создание условий для осознания учащимся необходимости нового способа своего предыдущего опыта, например, за счет создания «эффекта» невозможности разрешения ситуации в силу отсутствия в данный момент адекватных понятийных средств ее анализа;
2. **категоризация** – введение знаково-символического и визуального обозначения понятия с последующим постепенным увеличением степени обобщенности знаково-символического и визуального «языков» представления его содержания, ориентация ребенка на выявление частных и общих признаков соответствующего понятия;
3. **обогащение** – накопление и дифференциация опыта оперирования вводимым понятием, расширение возможных ракурсов осмысления его содержания;
4. **перенос** – применение усваиваемого понятия в разных ситуациях. В том числе и в условиях самостоятельного выстраивания отдельных аспектов его содержания;
5. **свертывание** – экстренная реорганизация всего множества имеющихся у ребенка сведений относительно данного понятия и превращение их в обобщенную знаниевую структуру. Развернутый на предыдущих фазах субъективный образ понятия на этой фазе должен быть представлен в сжатой, концентрированной форме.

Занятия в ТО «Системное программирование» проходят по системе развивающего обучения + проблемное обучение + компьютерное обучение. Лекционно-зачетная форма работы позволяет передать и обработать очень большой объем материала и практически отработать его на ЭВМ. Проблемное обучение позволяет приобрести прочные ЗУН, усвоить способы самостоятельной деятельности, развить познавательные и творческие способности учащихся. Усвоение знаний учащимися носит спонтанный характер, в процессе обучения ученик повторяет путь человечества в познании. Такой подход позволяет развивать мышление, научить думать, анализировать, делать выводы, но требует огромных затрат времени. Именно поэтому используется комбинированный метод обучения: лекции + развивающее + проблемное. Задачей учителя в рамках этой модели не просто сообщить некоторые факты, мнения, события, а поставить перед учениками проблему и предложить самостоятельно найти выход из нее.

Методические приемы при проведении практических работ, основанных на проблемном обучении:

1. учитель подводит учащихся к противоречию, предлагает ученикам самим найти способ его разрешения;
2. излагает различные точки зрения на одну проблему;
3. предлагает классу рассмотреть проблему с разных сторон, сравнить, обобщить, сделать выводы, сопоставить факты и мнения;
4. ставить проблемные теоретические и практические задания, например, исследовательские;
5. ставить проблемные задачи с недостатком или избытком информации;
6. задавать вопросы с заведомо допущенными ошибками.

Формы проведения занятий в основном три:

- лекция-беседа. Используется для теоретических знаний;
- работа в группах. Используется для практической отработки полученных знаний, работы над творческим проектом. Достоинства метода: налаживается взаимодействие, взаимопомощь между учащимися.
- индивидуальная работа. В основном используется на завершающих этапах обучения (3 год). Главная задача научить самостоятельной работе, самостоятельному поиску и обработке информации.

2.3. Ресурсное обеспечение

Для успешной реализации цели и задач, поставленных в данной образовательной программе, необходима соответствующая материально-техническая база. Компьютерный класс, в котором осуществляется обучение, имеет следующее оснащение: стационарные ПК, учебная доска.

На ПК установлены операционные системы Windows 7, 8, 10, стандартные и прикладные программы, способствующие подготовке грамотных специалистов в области пользовательских компьютерных знаний и программирования.

2.4. Оценочные материалы

Контроль полученных знаний – крайне важный момент в процессе обучения, он позволяет не только получить четкое представление об уровне полученных учащимися знаний, но и является обратной связью в системе учитель-ученик. Это позволяет внести коррективы в процесс обучения, изменить форму подачи материала, обратить особое внимание на «слабые» места в обучении. В данной системе предлагается использовать следующие формы контроля знаний:

- зачет, используется для контроля усвоения большого раздела, темы. Формы: письменный (опрос большого количества учащихся за минимальное время), устный (учитель опрашивает консультантов, а они учащихся своих групп. Учащиеся, слушая и рассказывая материал друг другу, как бы учат сами себя);
- практическая, лабораторная работа. Контроль умений и навыков работы за компьютером;
- компьютерный опрос, контрольная работа. Учащийся проходит тестирование в специальной программе, по системе вопросов, разработанных учителем;
- самостоятельное исследование программного продукта. Работа в программной среде, незнакомой ученику, но имеющей общие принципы работы с уже изученным программным обеспечением.

На первом, втором и третьем году обучения используются следующие методы контроля знаний:

- **контрольная работа.** Используется для проверки теоретических знаний;

- **лабораторная работа.** Используется для отработки теоретических знаний на практике;
- **творческая работа.** Используется для окончательного закрепления, полученных теоретических знаний за весь курс лекций, а так же самостоятельного изучения выбранной области программирования.

Для создания более благоприятного психологического климата проводятся:

- ставшие уже традиционными, новогодние огоньки,
- игра "Брейн-ринг для программистов". Участники делятся на две - три команды и в течение десяти - пятнадцати секунд придумывают ответ на вопрос из области программирования (строение ЭВМ, вопросы на сообразительность, развивающие вопросы). В качестве судьи, решающего какая из команд первой нажала кнопку, выступает компьютер.
- для учащихся ежегодно проводится олимпиада. На ней предлагается решить задачи различного уровня сложности: от предельно простой, до достаточно высокого уровня сложности. Проведение олимпиады имеет цель не только повысить уровень знаний участников, но и научить их концентрировать свои силы на поставленной задаче, научить их не стандартно и творчески мыслить, отстаивать свою точку зрения.
- Кружковцы принимают участие в городских, областных, Российских, Международных конкурсах и чемпионатах по программированию: научно-практическая конференция и конкурс творческих работ по программированию в ЦДТ «Солнечный», конференция им. А. А. Ухтомского, чемпионат по программированию ПИК, проводимый в Рыбинской государственной авиационной технологической академией. Учащиеся ТО «Системное программирование» являются призерами и победителями Всероссийских конференций: «Открытие» (г. Ярославль), «Юность, Наука, Культура» (г. Обнинск). Международных конференций: «Старт в науку» (г. Долгопрудный). Областных и Всероссийских олимпиад по информатике. В 2005 г. воспитанник творческого объединения стал призером Международной научной конференции школьников «Колмогоровские чтения» (г. Москва). Воспитанники ТО «Системное программирования» участвуют и побеждают в выше указанных профильных мероприятиях систематически. Участие в этих мероприятиях позволяет не только продемонстрировать свои силы, соотнести их с силами других программистов, но и получить новые знания и навыки. В 2006 г. учащемуся ТО «Системное программирование» Кузнецову Илье была назначена Государственная стипендия.

Контрольно-измерительные материалы

1. Критерий «Достижение цели проекта»	Балл	Оценка
Цель достигнута	3	
Цель достигнута частично	2	
Путь к цели только намечен	1	
Цели не достигли	0	
2. Критерий «Творческая самостоятельность проекта»		

Идея проекта оригинальная, яркая, неожиданная, предложена членами команды	3	
Идею проекта помог сформулировать учитель, члены команды ее разработали	2	
Идею проекта предложил учитель и совместно разрабатывал с командой на всех этапах	1	
Не смогли разработать оригинальную идею	0	
3. Критерий «Воплощение идеи»		
Идея проекта воплощена полностью, форма соответствует содержанию	3	
Идея проекта воплощена, но есть «шероховатости» в форме	2	
Идея проекта воплощена частично	1	
Идея проекта не нашла достойного воплощения	0	
4. Критерий «Умение работать в коллективе»		
В команде работали все, удавалось находить общий язык	3	
В команде работали по принуждению лидера, он сумел всех убедить	2	
Из группы «выпали» некоторые участники, но проект удалось реализовать	1	
Команда развалилась, проект не реализован	0	
5. Критерий «Качество презентации»		
Презентация выполнена качественно, на высоком уровне	3	
Презентация требует небольшой доработки	2	
Презентация очень слабая, не вызвала интереса у зрителей	1	

Презентация не подготовлена	0	
Общий балл максимально	15	

Лист самооценки учащихся «Мой проект»

№	Критерии и дескрипторы	Балл
1.	Зачем ты реализуешь проект?	
	Для решения собственной проблемы	3
	Для решения «чужой», но интересной и для меня проблемы	2
	Просто выполняю требование учителя	1
	Не знаю	0
2.	Какую позицию занимаешь в выработке проектного замысла?	
	Активно обсуждаешь, критикуешь, отбираешь и оцениваешь материал, выдвигаешь гипотезы по решению проблемы	3
	Участвуешь в обсуждении ситуации и проблемы, оцениваешь выдвинутую гипотезу, но пока не готов выдвинуть свою	2
	Наблюдаешь со стороны, но готов выполнить любые требования руководителя проекта	1
	Не участвовал (не хочу, не могу, не понимаю, неинтересно)	0
3.	Как ты думаешь, твой проект	
	направлен на решение культурной проблемы (самопознание, самоопределение, развитие культурной традиции и т.п.)?	3
	решает локальную задачу («улучшение конкретной ситуации»)?	2
	выполняется по заданному алгоритму («Что-то сделать по заданному образцу»)?	1

	Не определился с содержанием проекта	0
4.	Какие выбирались средства для реализации проектного замысла?	
	Удалось найти собственные средства для решения поставленной задачи	3
	Испытывали затруднение в выработке средств, обратились за помощью к учителю, в конце концов, разработали по аналогии с предложенными	2
	Средства для реализации замысла предложил учитель	1
	Не смог найти средства реализации проекта	0
5.	Как проработан твой проект содержательно?	
	Глубоко, качественно, эффективно проработано содержание проекта, привлечено максимальное количество информационных источников	3
	Содержание проекта в целом проработано, привлечено 2-3 информационных источника, указанных учителем	2
	Содержание проекта проработано в общих чертах, 1 информационный источник, указанный учителем	1
	Содержание не проработано	0
6.	Способен ли ты работать «в команде»?	
	Выступаешь инициатором и можешь создать «команду» для реализации задуманного; учишь интересы всех членов «команды»; умеешь слушать и слышать, договариваться	3
	Выступаешь инициатором и можешь создать «команду» для реализации своих замыслов; умеешь объяснить и поставить перед другими задачи	2
	Выполняешь задания учителя/ руководителя проекта, не обсуждая, не оценивая, не критикуя	1
	В обсуждении не участвовал, задания не выполнял	0
7.	Готов ли соблюдать проектную дисциплину?	

	Знаешь все этапы проектной деятельности и готов пошагово реализовывать проект от замысла до его воплощения и представления	3
	Проявил самостоятельность на всех этапах проектирования, кроме «промысливания и порождения новой идеи» (данный этап организовал учитель)	2
	Осуществлял деятельность по алгоритму, разработанному учителем и под руководством учителя	1
	Пока не смог организовать деятельность	0
8.	Каким видишь результат?	
	«Открыто», «изобретено», «сотворено» что-то новое (как следствие – «новое в себе»: новые знания, личностные качества, умения, открыл в себе способности), возник новый проект	3
	Замысел реализовал, «продукт» предъявил и получил хорошую оценку	2
	Участие в проектной деятельности принял, но в «продукте» нет ничего «твоего»/ не смог довести проект до завершения	1
	Не принял участия в проектной деятельности	0
	ВСЕГО	24
	Если ты набрал 24 - 23балла, то владеешь высоким уровнем проектной культуры, можешь самостоятельно задумывать и реализовывать проекты	24-23
	Если ты набрал от 22 до 16 баллов, то ты владеешь многими элементами проектной культуры, но пока не умеешь порождать проектные идеи	22-16
	Если ты набрал от 15 до 10 баллов, то ты делаешь первые шаги в проектировании	15-10

2.5. Воспитательный аспект

Дополнительное образование сегодня – это необходимое звено в воспитании многогранной личности, органично и естественно сочетающее в себе процессы обучения ребенка конкретной прикладной деятельности и разностороннего воспитания. Организации дополнительного образования обладают существенным воспитательным потенциалом и благоприятными условиями для поддержки творческих устремлений детей в самопознании, самоопределении, самореализации и самоутверждении.

В дополнительной общеобразовательной общеразвивающей программе «Начальное программное» поставлено решение следующих воспитательных задач:

✓ формировать трудолюбие, дисциплинированность, самостоятельность, активность, а также навыки культуры и бесконфликтного поведения.

Планируемый результат воспитания.

Результатом воспитания выступают личностные изменения человека, выражаемые в системе отношений к миру, к обществу и к самому себе:

- Готовность к продолжению образовательной деятельности;
- Наличие устойчивого познавательного интереса, стремление к самопознанию и творчеству;
- Наличие знаний, специальных умений и навыков, необходимых для успешного продолжения образования;
- Способность находить творческие решения учебных и практических задач в рамках дополнительной образовательной программы;
- Обладание эмоционально – волевыми качествами регуляции своего поведения в конфликтных ситуациях;
 - - готовность работать в команде;
 - - сформированы общие базовые ценности российского общества, активная гражданская позиция;
 - - сформированы нравственные качества личности.

Календарный план воспитательной работы

№ п/п	Название мероприятия	Дата проведения	Место проведения
1	Инструктажи о правилах поведения и технике безопасности	согласно графику	Центр «Солнечный»
2	Участие в профильных мероприятиях разного уровня: <ul style="list-style-type: none"> ➤ Муниципальный конкурс научно-технического творчества "Компьютерный гений" "Научно-практическая конференция , Пользователь ПК, Конкурс компьютерной графики ➤ Научная межмуниципальная конференция школьников, посвященной памяти академика А.А. Ухтомского ➤ Всероссийская конференция студентов и школьников "Сту- 	согласно графику	Центр «Солнечный», интернет-пространство, выезды в другие города, области

	<p>пень в науку"</p> <ul style="list-style-type: none"> ➤ Российская научная конференция школьников «Открытие» ➤ Турнир по программированию НПО «Криста» и ЯрГУ им. Демидова для школьников и студентов колледжей «#Develobear» ➤ Всероссийский конкурс "Творчество без границ" Компьютерные игровые программы. 		
3	<p>Участие в создании поздравлений к праздникам:</p> <ul style="list-style-type: none"> ➤ День учителя ➤ День матери ➤ Новый год ➤ 23 февраля ➤ 8 марта ➤ 9 мая 	согласно календарю	Центр «Солнечный»
4	<p>Участие в общих воспитательных мероприятиях Центра:</p> <ul style="list-style-type: none"> ➤ День открытых дверей ➤ Онлайн - опросы для школьников и молодежи ➤ Посвящение в кружковцы ➤ Новогодние праздники ➤ Благотворительные акции, организованные Рыбинской Епархией 	согласно календарю	Центр «Солнечный», другие организации и ведомства
5	<p>Проведение в творческом объединении воспитательных мероприятий:</p> <ul style="list-style-type: none"> ➤ День учителя ➤ День матери ➤ Новый год ➤ 23 февраля ➤ 8 марта ➤ 9 мая ➤ Итоговые праздники в группах ➤ Посещение кинотеатров ➤ Посещение городского сквера 	согласно календарю	Центр «Солнечный»
6	<p>Организация выставок работ учащихся:</p> <ul style="list-style-type: none"> ➤ Рисунки компьютерной графики 	в течение года	Центр «Солнечный», интернет-пространство
	Открытые занятия:	согласно плана рабо-	Центр «Солнечный»

	<ul style="list-style-type: none"> ➤ Профессия «Программист» ➤ Современное занятие с элементами ИКТ. Развитие логического мышления ➤ Формирование профессионально - важных качеств у обучающихся ➤ Воспитание культуры умственного труда ➤ Профессиограмма "Программист" 	ты	
	<p>Информационные занятия:</p> <ul style="list-style-type: none"> ➤ Мы наследники Победы ➤ Снятию блокады Ленинграда ➤ Вывод советских войск из Афганистана, ➤ День защитника Отечества ➤ День героев Отечества ➤ и другие 	согласно календарю	Центр «Солнечный», интернет-пространство

3. Список источников

1. Абель Питер Язык Ассемблера для IBM PC и программирования. – М.: Высшая школа, 1992.
2. Абрамов С.А. Задачи по программированию. – М.: Наука, 1988.
3. Аргов Д.И. Объектно-ориентированное программирование: новый взгляд и новое мышление – Рыбинск, 2005.
4. Аргов Д.И. Сборник задач и алгоритмов для олимпиадной подготовки по программированию. – Рыбинск, 2003, 2004.
5. Аргов Д.И. Справочное пособие по языку ассемблер. Основные принципы функционирования сетей ЭВМ. – Рыбинск, 2006.
6. Бадин Н.М., Волченков С.Г. Ярославские олимпиады по информатике. – Ярославль, 1995.
7. Гигиенические требования к видеодисплейным терминалам, персональным электронно-вычислительным машинам и организации работы. Санитарные правила и нормы. СанПин 2.2.2/2.4.1340-03. – М., 2003.
8. Закон об образовании от 22.08.2004 г. № 122-ФЗ.
9. Зубков С.В. Язык Ассемблер. – М.: 1998.
10. Иванов Б.Н. Дискретная математика. Алгоритмы и программы – М.: Лаборатория Базовых Знаний, 2002.
11. Окулов С.М. 100 задач по информатике – Киров, 2000.
12. Пильщиков В.Н. Сборник упражнений по языку Паскаль – М.: Наука, 1989.
13. Пинаев В.Н. Теория и практика творческих соревнований по информатике– Рыбинск, 2000.
14. Усакова О.Ф. Программирование на языке Паскаль – СПб: Питер, 2002.
15. Учебное пособие для студентов педагогических институтов /Под ред. Бабанского Ю.К. – М., 1989
16. Учебное пособие для студентов высших учебных заведений /Под ред. Лебедева О.Е. –М., 2000.
17. Холодная М.А. Психология интеллекта .Парадоксы исследования. – СПб.: Питер, 2002.

Нормативно-правовые документы

1. Федеральный закон от 29.12.2012 г. №273-ФЗ «Об образовании в РФ»
2. Федеральный закон от 31.07.2020 № 304-ФЗ «О внесении изменений в Федеральный закон «Об образовании в РФ» по вопросам воспитания обучающихся»
3. Приказ Министерства просвещения России от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
4. Приказ Министерства просвещения РФ от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем ДОД»
5. Концепция развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства РФ от 31 марта 2022 г. №678-р)
6. Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства РФ от 29 мая 2015 г. № 996-р)
7. Профессиональный стандарт «Педагог дополнительного образования детей и взрослых» (Приказ Минтруда России от 22.09.2021 N 652н)
8. «Об утверждении Порядок организации и осуществления образовательной деятельности при сетевой форме реализации образовательных программ (Приказ Министерства науки и высшего образования РФ и Министерства просвещения РФ от 05.08.2020 №882/391)

9. Постановление Главного государственного санитарного врача РФ от 28.09.2020 №28 «Об утверждении СанПиН 2.4.4.3648-20 «Санитарно-эпидемиологические требования к организации воспитания и обучения, отдыха и оздоровления детей и молодежи»