

Департамент образования  
администрации городского округа Тольятти

Муниципальное бюджетное образовательное учреждение  
дополнительного образования «Центр Гранит»  
городского округа Тольятти

Принята на основании  
решения педагогического совета  
протокол № 4  
от "09" июня 2021 г.



Утверждаю

Директор

МБОУДО «Центр Гранит»

П.А. Завьялов

приказ № 147/1-ОД от "09" июня 2021 г.

Дополнительная общеобразовательная общеразвивающая программа

**«Классическое структурное  
программирование2»**

*технической направленности*

Срок реализации 3 года

Возраст учащихся 10-16 лет

Составитель:

Батижевская С.Л. педагог  
дополнительного образования

Тольятти, 2021

## Пояснительная записка

Человечество всем стадиям своего развития дало меткие эпитеты: каменный век; бронзовый век; индустриальный век; космический век. Наступивший XXI век уже получил свое название – компьютерный, информационный век. Информация, информированность, информатизация, информатика... Все эти термины прочно вошли в словарный запас наших современников. Современная деловая жизнь предъявляет строгие требования к молодому поколению, пополняющему ряды работников всех сфер деятельности. Начинаящий работник должен обладать достаточным багажом знаний и умений, чтобы не быть невостребованным на рынке труда, чтобы быть социально защищенным в нашем скоростном, информационно насыщенном мире.

Дополнительная общеобразовательная общеразвивающая программа «Классическое структурное программирование» технической направленности, предусматривает освоение основ алгоритмизации и программирования, освоение математической логики; свободное владение вычислительной техникой. Программа разработана согласно Федеральному закону от 29.12.2012 г. №273-ФЗ «Об образовании в Российской Федерации»; Концепции развития дополнительного образования в РФ (утверждена распоряжением Правительства РФ от 04.09.2014 № 1726-Р); Стратегии развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства Российской Федерации от 29 мая 2015 г. № 996-р; Приказу Министерства просвещения России от 9.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»; Постановлению Главного государственного санитарного врача Российской Федерации от 4 июля 2014 года № 41 «Об утверждении [СанПиН 2.4.4.3172-14](#) "Санитарно-эпидемиологические требования к устройству, содержанию и

организации режима работы образовательных организаций дополнительного образования детей»; Приказу министерства образования и науки Самарской области от 20.08.2019 г. № 262-од «Об утверждении Правил персонифицированного финансирования дополнительного образования детей в Самарской области на основе сертификата персонифицированного финансирования дополнительного образования детей, обучающихся по дополнительным общеобразовательным программам»; Методическим рекомендациям по проектированию дополнительных общеразвивающих программ, направленных письмом Минобрнауки России от 18.11.2015 № 09-3242; «Методическим рекомендациям по разработке дополнительных общеобразовательных программ» (Приложение к письму министерства образования и науки Самарской области 03.09.2015 № МО -16-09-01/826-ТУ).

### **Актуальность**

А. Берг писал: «Научимся программировать – научимся обучать». Это актуально как для педагога, ищущего пути оптимизации учебного процесса, так и для учащегося, осознавшего важность самообучения, самообразования. Компьютерные технологии развивают идеи программированного обучения и открывают совершенно новые технологические варианты обучения, связанные с уникальными возможностями компьютеров и телекоммуникаций. Широкое освоение этих технологий в практике образования возможно лишь при безусловной компьютерной грамотности, которую можно рассматривать как особую часть содержания компьютерной технологии. Таким образом, разработка методики преподавания информатики – одна из актуальных проблем современной педагогики (и в общедидактическом и частнометодическом планах).

**Новизна** программа состоит в том, что она разработана с учётом современных тенденций в образовании по принципу блочно-модульного освоения материала, что максимально отвечает запросу социума на возможность выстраивания ребёнком индивидуальной образовательной траектории.

**Отличительной особенностью** программы является применение конвергентного подхода, предусматривающего переориентации учебной деятельности с познавательной на проектно-конструкторскую.

**Педагогическая целесообразность** предлагаемой программы заключается в использовании метода проектов с первого занятия и до последнего. Каждое занятие посвящено какой-либо теме из классического набора, но цель использования того или иного понятия у каждого учащегося своя собственная. Таким образом, переплетаются методы проектов с индивидуальным обучением, применением на занятиях деятельностного подхода в организации обучения, где ребенок, действуя сам, ответственно относится к выполнению порученного задания. В программу включены коллективные практические занятия, развивающие коммуникативные навыки и способность работать в команде. Практические занятия помогают развивать у детей воображение, внимание, абстрактное мышление, умение свободно выражать свои чувства и настроения, работать в коллективе. Программирование способствует формированию умения учиться, добиваться результата, создавать индивидуальный творческий продукт, получать новые знания. Целесообразность предложенной программы также состоит в том, что она сориентирована, начиная с первого занятия, не только на развитие стандартных пользовательских навыков, но и на техническое творчество учащихся. Базой для такого подхода с первого года обучения становятся профессиональные программные продукты (Microsoft Office, Borland Pascal, Adem и т.д.), которые органично переплетаются с темами по программированию. Программа разработана с учётом современных

тенденций в образовании по принципу блочно-модульного освоения материала, что максимально отвечает запросу населения на возможность выстраивания учащимся индивидуальной образовательной траектории.

**Цель программы:** Создание условий для освоения учащимися фундаментальных понятий современной информатики, овладения навыками работы с аппаратными и программными средствами современных персональных компьютеров.

Задачи:

*Обучающие:*

- научить пользоваться вычислительной техникой от простого пользователя до программирования;
- помочь освоить основы алгоритмизации и программирования;
- ознакомить элементами математической логики;

*Развивающие:*

- развивать техническое творчество;
- развивать внимание, логику;

*Воспитательные:*

- воспитать целеустремленность, трудолюбие;
- чувство товарищества, взаимопомощи.

**Возраст учащихся, участвующих в реализации программы**

Программа рассчитана на учащихся в возрасте 10 – 16 лет.

**Сроки реализации:** программа рассчитана на 3 года, общий объем - 324 часа. 108 часов в год, 3 часа в неделю.

**Режим занятий** Занятие проводится 1 раз в неделю по 3 часа, 2 раз в неделю, 1 раз - 2 часа и один раз - 1 час, 3 раза в неделю по - 1 час. Продолжительность каждого занятия – 45 минут. Перерыв между занятиями 10 минут.

Продолжительность каждого занятия – 45 минут. Перерыв между занятиями 10 минут.

**Наполняемость учебных групп:** составляет от 10 до 15 человек, исходя из оснащения техникой.

### **Формы организации деятельности**

Форма организации деятельности по программе – групповая, индивидуальная.

### **Формы и методы обучения**

Форма обучения: очная.

- занятие;
- лекция;
- практическая работа;
- защита проекта.

Методы обучения

- Словесные методы: рассказ, беседа, сообщение.
- Наглядные методы: презентация, демонстрация рисунков, плакатов, иллюстраций.
- Практические методы: выполнение практических работ.

Сочетание словесного и наглядного методов обучения, позволяют адаптировать учащегося к восприятию материала, направить его на

Ожидаемые результаты:

*Личностные:*

- гражданская идентичность учащихся;
- чувство любви к родной стране, к её природе, культуре, с интересом к её истории, уважительном отношении к другим странам, народам, их традициям;

- чувство гордости за свою страну и малую родину;
- формирование уважительного отношения к иному мнению, истории и культуре своего народа и других народов, живущих рядом; выработка умения терпимо относиться к людям иной национальной принадлежности;
- положительное отношение к процессу учения, к приобретению знаний и умений, стремление преодолевать возникающие затруднения;
- наличие мотивации к творческому труду и бережному отношению к материальным и духовным ценностям, формирование установки на безопасный, здоровый образ жизни;
- начальные навыки саморегуляции;
- осознанность в отношении к себе как к индивидуальности и, одновременно, как к члену общества с ориентацией на проявление доброго отношения к людям, уважения к их труду, на участие в совместных делах, на помощь людям, в том числе сверстникам.

Метапредметные:

*Познавательные:*

- анализировать информацию;
- преобразовывать познавательную задачу в практическую;
- выделять главное, осуществлять выбор наиболее эффективных способов решения;
- прогнозировать результат.

*Регулятивные:*

- планировать, контролировать и оценивать учебные действия в соответствии с поставленной задачей и условием её реализации в процессе познания;
- понимать причины успеха/неуспеха учебной деятельности;
- конструктивно действовать даже в ситуациях не успеха;

- самостоятельно учитывать выделенные педагогом ориентиры действия в новом материале;
- вносить коррективы в действие после его завершения на основе оценки и учета характера сделанных ошибок;
- адекватно воспринимать предложения и оценку педагогов, товарищей и родителей;
- готовность оценивать свой труд, принимать оценки одноклассников, педагогов, родителей.

*Коммуникативные:*

- учитывать разные мнения и интересы и обосновывать свою позицию;
- приходить к общему решению в совместной работе (сотрудничать с одноклассниками);
- сотрудничать со взрослыми и сверстниками в разных социальных ситуациях;
- не создавать конфликтов и находить выходы из спорных ситуаций

Предметные результаты.

Модульный принцип построения программы предполагает описание предметных результатов в каждом конкретном модуле.

В разработанной программе все темы (теоретические и практические) слиты воедино, изучение материала представляет собой непрерывный процесс, по завершении которого к учащемуся предъявляются следующие требования по предмету:

- Свободное владение вычислительной техникой;
  - знание основных устройств ВТ;
  - знание классификации программных средств;
  - умение работать с сервисными программами (оболочками, архиваторами и т.д.)
  - умение работать в редакторах (текстовых и графических),



электронных таблицах, интегрированных средах и т.д.;

- Знание основ алгоритмизации и программирования:
  - знание основных типов переменных;
  - знание основных конструкций алгоритмического языка (линейное следование, ветвление, цикл);
- Знание элементов математической логики;
- Умения и навыки:
  - умение анализировать поставленную задачу;
  - умение выбирать методы и способы решения поставленной задачи;
  - умение анализировать полученные результаты и делать выводы.

Кроме наличия всех вышеперечисленных знаний, умений и навыков учащийся должен обладать стройным представлением целостной картины информационного мира, высокой компетентностью в области прикладной информатики.

Разработанная программа позволяет динамично реагировать на изменение требований, предъявляемых к аппаратному и программному обеспечению. Данная программа не привязана ни к какому конкретному дидактическому наполнению курса. Выбор конкретных задач зависит от ориентации каждой конкретной группы учащихся. Кроме тематической ориентации каждая предлагаемая учащимся задача предполагает различную степень детализации в зависимости от уровня развития способностей каждого учащегося, что позволяет осуществлять дифференцированный и индивидуальный подход. В программе отведено достаточное время для проектной работы, позволяющей раскрыть все творческие способности учащегося. Но с увеличением обучения, одновременно увеличивается и без того не малая нагрузка (физическая, эмоциональная и интеллектуальная) на ученика, на неокрепший организм ребенка. Необходимо, не сокращая объема получаемых знаний и умений, уменьшить загруженность учащихся (особенно учащихся младшего и среднего звена общеобразовательной школы).

Для решения поставленной задачи первоначально автором были подготовлены тексты – "шпаргалки".

**Преподавание информатики интегрировано со следующими дисциплинами:**

В программе отведено достаточное время для проектной работы, позволяющей раскрыть все творческие способности учащегося.

Проект представляет собой самостоятельную работу учащегося, рассчитанную на продолжительное время. Учащийся самостоятельно выбирает предметную область и тему своего проекта, формулирует цель работы, выдвигает предположение, выбирает средства и методы решения поставленной перед собой задачи, реализует решение, анализирует полученные результаты и делает выводы о достижении поставленной цели.

При изучении курса «Информатика» предусмотрены следующие формы контроля знаний: терминологический диктант, самостоятельная работа, контрольная работа, зачет по теме, защита проекта.

**Учебный план**

№ модуля	Название модуля	Количество часов		
		Всего	Теория	Практика
1	Основы структурного программирования	108		
2	Основные алгоритмические структуры	108		
3	Программирование в алгоритмах	108		
<b>ИТОГО</b>		<b>324</b>		

**Критерии оценки знаний, умений и навыков**

**при освоении программы**

Для того чтобы оценить усвоение программы, в течение года используются следующие методы диагностики: собеседование,

наблюдение, анкетирование, выполнение отдельных творческих заданий, проектная деятельность, практическая работа, тестирование, участие в конкурсах.

По завершению учебного плана каждого модуля оценивание знаний проводится по результатам презентации проекта, интерактивного занятия.

Применяется 3-х балльная система оценки знаний, умений и навыков обучающихся (выделяется три уровня: ниже среднего, средний, выше среднего). Итоговая оценка результативности освоения программы проводится путём вычисления среднего показателя, основываясь на суммарной составляющей по итогам освоения каждого модуля.

Уровень освоения программы ниже среднего – ребёнок овладел менее чем 55% предусмотренных знаний, умений и навыков, испытывает серьёзные затруднения при работе с учебным материалом; в состоянии выполнять лишь простейшие практические задания педагога.

Средний уровень освоения программы – объём усвоенных знаний, приобретённых умений и навыков составляет 55-75%; работает с учебным материалом с помощью педагога; в основном, выполняет задания на основе образца; удовлетворительно владеет теоретической информацией по темам курса, умеет пользоваться литературой.

Уровень освоения программы выше среднего – учащийся овладел на 75-100% предусмотренным программой учебным планом; работает с учебными материалами самостоятельно, не испытывает особых трудностей; выполняет практические задания с элементами творчества; свободно владеет теоретической информацией по курсу, умеет анализировать литературные источники, применять полученную информацию на практике.

**Формы контроля качества образовательного процесса:**

- собеседование,
- наблюдение,
- интерактивное занятие;
- анкетирование,
- выполнение творческих заданий,
- защита проекта, тестирование,
- участие в конкурсах в течение года.

### **Модуль первый «Основы структурного программирования»**

**Цель:** Овладение учащимися начальными знаниями программирования.

#### **Задачи:**

*обучающие:*

- ознакомить детей с основами структурного программирования;
- научить работать с основными алгоритмическими конструкциями;
- научить основам программирования.

*развивающие:*

- развить логическое мышление ребенка;
- формировать навыки программирования;

*воспитательные:*

- воспитать усидчивость, дисциплинированность, доброжелательность к сверстникам.

По окончании модуля учащиеся будут знать:

- правила разработки программ;
- ознакомятся с алгоритмами линейными, разветвляющимися, циклами.

уметь:

- пользоваться языком программирования;
- создавать простейшие программы.

**Учебно-тематический план**  
**Первый год обучения (Модуль 1)**

№ п/п	Тема	Количество часов			Формы контроля/аттестации
		Теория	Практика	Всего	
<b>Раздел 1. Программирование: история и современность</b>					
1.	Тема 1.1. Введение в курс «Классическое структурное программирование». Инструктаж	3	1	4	Собеседование, наблюдение, самостоятельная работа
2.	Тема 1.2 Операции, операнды, операторы	2	3	5	
3.	Тема 1.3 Уровни программного обеспечения	1	2	3	
<b>Раздел 2. Обще пользовательские сведения</b>					
4.	Тема 2.1. Оболочка к операционной системе	2	2	4	наблюдение, самостоятельная работа
5.	Тема 2.2. Текстовый редактор	2	4	6	
<b>Раздел 3. Основы математической логики</b>					
6.	Тема 3.1. Логические переменные и логические операции	2	6	8	наблюдение
<b>Раздел 4. Основы структурного программирования</b>					
7.	Тема 4.1. Переменные, операторы	2	4	6	наблюдение, самостоятельная работа
8.	Тема 4.2. Алгоритмы	2	4	6	
9.	Тема 4.3. Блочное программирование	2	3	5	
10.	Тема 4.4. Промежуточный контроль	0	1	1	
11.	Тема 4.5. Линейные операторы	2	4	6	
12.	Тема 4.6. Операторы условия	2	4	6	
13.	Тема 4.7. Операторы цикла повторения	1	2	3	
14.	Тема 4.8. Подпрограммы	2	4	6	
15.	Тема 4.9. Создание программного обеспечения	2	4	6	
16.	Тема 4.10. Файлы	2	4	6	
17.	Тема 4.11. Создание прикладных программных продуктов	3	10	13	
<b>Раздел 5. Ученик - пользователь компьютера</b>					
18.	Тема 5.1. Интегрированные системы	2	4	6	наблюдение, самостоятельная работа
19.	Тема 5.2. Пакеты прикладных программ (ППП)	2	4	6	
20.	Тема 5.3. Подведение итогов изучения курса «Классическое структурное программирование» 1-	0	5	5	

	го года обучения выполнение творческого контрольного задания				
<b>ИТОГО</b>		<b>36</b>	<b>72</b>	<b>108</b>	

## **Второй модуль «Основные алгоритмические структуры»**

**Цель:** Создание условий для формирования навыков программирования.

### **Задачи:**

*обучающие:*

- помочь освоить язык программирования;
- обучить основам программирования;
- научить создавать прикладные программные продукты.

*развивающие:*

- развить умение осмысливать собственные действия;
- формировать навыки программирования.

*воспитательные:*

- воспитывать усидчивость, целеустремленность, чувство взаимопомощи.

По окончании модуля учащиеся будут знать:

- виды алгоритмов;
- способы записи алгоритмов;

уметь:

- записывать алгоритмы на языке программирования;
- решать задачи с использованием операторов: линейных, условия, цикла.

## **Учебно-тематический план**

### **Второй год обучения (Модуль 2)**

№ п/п	Тема	Количество часов			
		Теория	Практика	Всего	
Раздел 1. Программирование: история и современность					
1.	Тема 1.1 Введение в курс «Классическое структурное программирование» 2-го года	1	0	1	Собеседование

	обучения				
2.	Тема 1.2 Операции, операнды, операторы. Целочисленная арифметика	2	6	8	
3.	Тема 1.3 Уровни программного обеспечения	2	0	2	
<b>Раздел 2. Общепользовательские сведения</b>					
4.	Тема 2.1. Оболочка к операционной системе: повторение	2	5	7	наблюдение
5.	Тема 2.2. Промежуточный контроль	0	1	1	
6.	Тема 2.3. Текстовый редактор. Дополнительные возможности	3	8	11	
<b>Раздел 3. Основы математической логики</b>					
7.	Тема 3.1. Логические переменные и логические операции. Построение сложных условий	5	7	12	самостоятельная работа
<b>Раздел 4. Основы структурного программирования</b>					
8.	Тема 4.1. Переменные, операторы. Основные назначения в программе	2	10	12	Собеседование, самостоятельная работа
9.	Тема 4.2. Алгоритмы. Основные алгоритмические структуры	3	6	9	
10.	Тема 4.3. Блочное программирование. Виды	2	7	9	
11.	Тема 4.4. Линейные операторы. Повторение	0	1	1	
12.	Тема 4.5. Операторы условия. Полное/неполное условие. Способы задания и описания	2	3	5	
13.	Тема 4.6. Операторы цикла: решение задач повышенного уровня с различными видами циклов	2	10	12	
14.	Тема 4.7. Подпрограммы: решение задач повышенного уровня	1	5	6	
15.	Тема 4.8. Создание прикладных программных продуктов	1	9	10	
16.	Тема 4.9. Подведение итогов изучения курса «Классическое структурное программирование» 2-го года обучения	0	2	2	
<b>ИТОГО</b>		<b>28</b>	<b>80</b>	<b>108</b>	

### **Третий модуль «Программирование в алгоритмах»**

**Цель:** Создание условий для овладения языком программирования.

**Задачи:**

*обучающие:*

- помочь освоить язык программирования;
- обучить алгоритмам некоторых популярных задач;
- научить составлять различные алгоритмы на языке программирования.

*развивающие:*

- развить умение осмысливать собственные действия при составлении программ;
- формировать навыки программирования.

*воспитательные:*

- воспитывать усидчивость, целеустремленность, чувство взаимопомощи.

По окончании модуля учащиеся будут знать:

- комбинаторные алгоритмы, алгоритмы перебора, алгоритмы на графах, алгоритмы вычислительной геометрии;
- как работать с многоуровневыми числами;

уметь:

- создавать прикладные программные продукты.
- решать задачи на языке программирования.

### Учебно-тематический план

#### Третий год обучения (Модуль 3)

№ п/п	Тема	Количество часов			Формы контроля/аттестации
		Теория	Практика	Всего	
Раздел 1. Программирование: история и современность					
17.	Тема 1.1 Введение в курс «Классическое структурное программирование» 3-го года обучения	1	0	1	Собеседование, наблюдение, самостоятельная работа
18.	Тема 1.2 Операции, операнды, операторы. Целочисленная арифметика	2	6	8	
19.	Тема 1.3 Уровни программного обеспечения	2	0	2	
Раздел 2. Общепользовательские сведения					
20.	Тема 2.1. Оболочка к операционной системе: повторение	2	5	7	наблюдение, самостоятельная
21.	Тема 2.2. Промежуточный контроль	0	1	1	



22.	Тема 2.3. Текстовый редактор. Дополнительные возможности	3	8	11	работа
Раздел 3. Основы математической логики					
23.	Тема 3.1. Логические переменные и логические операции. Построение сложных условий	5	7	12	наблюдение,
Раздел 4. Основы структурного программирования					
24.	Тема 4.1. Переменные, операторы. Основные назначения в программе	2	10	12	Собеседование, самостоятельная работа
25.	Тема 4.2. Алгоритмы. Основные алгоритмические структуры. Повторение	3	6	9	
26.	Тема 4.3. Арифметика многоразрядных целых чисел	2	7	9	
27.	Тема 4.4. Комбинаторные алгоритмы	-	1	1	
28.	Тема 4.5. Перебор и методы его сокращения	2	3	5	
29.	Тема 4.6. Алгоритмы на графах	2	10	12	
30.	Тема 4.7. Алгоритмы вычислительной геометрии	1	5	6	
31.	Тема 4.8. Создание прикладных программных продуктов	1	9	10	
32.	Тема 4.9. Подведение итогов изучения курса «Классическое структурное программирование» Защита проекта	0	2	2	
<b>ИТОГО</b>		<b>28</b>	<b>80</b>	<b>108</b>	

### Содержание программы

Тема 1.1. Введение в курс «Классическое структурное программирование», Инструктаж.

История и современность.

Понятия: информация, информатика, программирование, программное обеспечение. Системы счисления: "10", "12", "16", "2" и др. - их историческое происхождение, следы этих систем в языке и байте. Двоичная система счисления, как система кодирования в ЭВМ.

Тема 2. Операции, операнды, операторы.

Понятия: константы, переменные, выражения, операции, операнды, операторы, функции, команды, программы (программное обеспечение).

Тема 3 Уровни программного обеспечения

3Уровни программного обеспечения:

- операционные системы;
- оболочки к операционным системам;
- прикладное программное обеспечение.
- Иерархическая (древесная) структура программного обеспечения

## **II. Общепользовательские сведения**

### 1. Оболочка к операционной системе

Понятия: директория (каталог), дерево, файл (текстовый, командный, выполняемый), имя файла, расширение имени файла, создание, изменение, сохранение, копирование, удаление файлов и директорий.

### 2. Текстовый редактор

Понятия: Создание и редактирование текста, курсор, вставка, замена, копирование, форматирование абзаца, вывод текста на печать, запись текстового файла в директорию ученика.

## **III. Основы математической логики.**

### 1. Логические переменные и логические операции.

Понятия: логические переменные, логические операции, логические выражения. Логические операции: AND, OR, NOT. Наборы значений логических переменных. Таблицы истинности.

## **IV. Основы структурного программирования.**

### 1. Переменные, операторы.

Типы переменных: простые (оригинальные) и сложные. Размерность типов. Области применения различных типов переменных (Boolean, Byte, Integer, Real, Char, String).

Типы операторов: линейные (ввода/вывода, чтения/записи, присваивания и т.д.), условные операторы (выбора), операторы цикла (повторения).

### 2. Алгоритмы.

Понятия: постановка задачи, алгоритм решения поставленной задачи, результат, оценка правильности алгоритма, единственность алгоритма, реализация алгоритма, виды алгоритмов.

### 3. Блочное программирование.

Понятие структурный блок. Виды структурных блоков. Пять принципов структурного программирования.

#### 4. Линейные операторы.

Линейные операторы: ввода/вывода, чтения/записи, присваивания.

#### 5. Операторы условия.

Понятия: IF - форматы записи, организация блоков BEGIN..END.

#### 6. Операторы цикла (повторения).

Операторы цикла (повторения) с постусловием (REPEAT) - синтаксис, область применения.

#### 7. Создание программного обеспечения.

Разработка алгоритма, создание блок-схемы, комментарии к программному обеспечению.

Написание программы - основы.

Отладка программного обеспечения. Принципы отладочных печати и подсветки.

Тестирование готового программного продукта.

#### 1. Сложные типы переменных массив (ARRAY), запись (RECORD).

Размерность типов. Области применения различных типов переменных.

Типы операторов: линейные (ввода/вывода, чтения/записи, присваивания и т.д.), условные операторы (выбора), операторы цикла (повторения).

2. Алгоритмы. Вспомогательные алгоритмы, метод пошаговой детализации. Понятия: постановка задачи, алгоритм решения поставленной задачи, результат, оценка правильности алгоритма, единственность алгоритма, реализация алгоритма, вспомогательный алгоритм, детализация, укрупнение.

#### 3. Блочное программирование.

Инвариантность к языкам принципов структурного программирования.

Анализ реализации в языках программирования различных структурных единиц. Способы организации структурных единиц в случае их отсутствия в языке. Сложные многоуровневые блок-схемы. Принцип вложенности структурных блоков.

#### 4. Операторы условия (выбора).

Операторы условия (выбора): IF, CASE - сравнительная характеристика реализации операторов условия (выбора) в различных языках программирования.

#### 5. Операторы цикла (повторения).

Операторы цикла (повторения) с заведомо известным количеством

повторений (FOR) - синтаксис, область применения, ограничения в различных языках программирования. Оператор цикла FOR для работы с матрицами.

Операторы цикла (повторения) с предусловием (WHILE) - синтаксис, область применения, ограничения в различных языках программирования. Оператор цикла WHILE для работы с массивами типа RECORD.

Операторы цикла (повторения) с постусловием (REPEAT) - синтаксис, область применения, ограничения в различных языках программирования. Оператор цикла REPEAT для работы с массивами типа RECORD.

Прочие применения операторов цикла.

## 5. Подпрограммы.

Процедуры и функции. Формальные и фактические параметры. Библиотеки стандартных процедур и функций.

## 7. Создание программного обеспечения.

Этапы создания программного продукта. Работа с экраном. Различные стили оформления экрана. Контроль корректного ввода/вывода. Оформление результатов выполнения программы: сообщения, таблицы, диаграммы, графики. Тестирование готового программного продукта. Защита и реклама.

### 1. Файлы.

Работа с файлами. Типизованные и не типизованные файлы. Файловые переменные. Чтение с диска и запись на диск. Процедуры и функции для обработки файлов.

### 2. Создание прикладных программных продуктов имеющих сложное структурное строение с использованием всех известных конструкций, отладка, тестирование, защита программного обеспечения,

Библиотеки стандартных процедур и функций, подключаемые модули, создание собственных библиотек.

Создание прикладных программных продуктов имеющих сложное структурное строение с использованием всех известных конструкций, отладка, тестирование, защита программного обеспечения.

Арифметика многоразрядных целых чисел. Основные арифметические

операции. Комбинаторные алгоритмы. Перестановки. Размещения. Сочетания. Скобочные последовательности. Представление графа в памяти компьютера. Поиск в графе. Алгоритм Дейкстры. Алгоритм Флойда. Алгоритмы вычислительной геометрии. Базовые процедуры. Задачи о прямоугольниках. Тестирование программы при решении задач.

#### **IV. Ученик - пользователь компьютера.**

##### **1. Интегрированные системы.**

Исследовательская деятельность. Работа в неизвестных средах (русифицированных и англоязычных). Самостоятельное изучение второго языка программирования. Сравнительный анализ программных продуктов одного класса.

##### **2. Пакеты прикладных программ (ППП).**

Многообразие ППП для различных областей деятельности. Принципы построения ППП. Интерфейс, "хелпирование", основные функции. Внедрение и сопровождение прикладных пакетов программ.

### **Обеспечение программы**

#### ***Методическое обеспечение***

##### Основные принципы, положенные в основу программы:

- принцип доступности, учитывающий индивидуальные особенности каждого
- ребенка, создание благоприятных условий для их развития;
- принцип демократичности, предполагающий сотрудничество педагога и учащегося;
- принцип системности и последовательности – знание в программе даются в определенной системе, накапливая запас знаний, дети могут применять их на практике.

##### Методы работы:

- *словесные методы:* рассказ, беседа, сообщения – эти методы способствуют обогащению теоретических знаний детей, являются источником новой информации;

- *наглядные методы*: презентации. Наглядные методы дают возможность более детального обследования объектов, дополняют словесные методы, способствуют развитию мышления детей. «Чем более органов наших чувств принимает участие в восприятии какого-нибудь впечатления или группы впечатлений, тем прочнее ложатся эти впечатления в нашу механическую, нервную память, вернее сохраняются ею и легче, потом вспоминаются»(К.Д. Ушинский);
- *практические методы*: выполнение творческих заданий. Данные методы позволяют воплотить теоретические знания на практике, способствуют развитию навыков и умений детей. Большое значение приобретает выполнение правил культуры труда, бережного отношения к технике, приспособлениям и материалам.

Сочетание словесного и наглядного методов учебно-воспитательной деятельности, воплощённых в форме рассказа, беседы, творческого задания, позволяют психологически адаптировать ребёнка к восприятию материала, направить его потенциал на познание истории языка программирования, расширению кругозора.

Занятие состоит из следующих структурных компонентов:

1. Организационный момент, характеризующийся подготовкой учащихся к занятию;
2. Повторение материала, изученного на предыдущем занятии;
3. Постановка цели занятия перед учащимися;
4. Изложение нового материала;
5. Практическая работа;
6. Обобщение материала, изученного в ходе занятия;
7. Подведение итогов;
8. Приведение в порядок рабочего места.

**Материально-техническое обеспечение программы**

Требования к помещению, в котором должны проводиться занятия:  
учебный класс.

Перечень оборудования и материалов, необходимых для занятий  
(на группу детей):

№ п/п	Наименование	
<b>Оборудование</b>		
1.	Ноутбуки	
2.	Проектор	
3	Колонки	
4	Доска	
5.	Принтер	
<b>Канцелярские принадлежности</b>		
1.	Карандаши цветные	
4.	Карандаши простые	
4.	Ножницы	
5.	Линейка	
<b>Материалы</b>		
1.	Бумага белая формата А-4	

## Литература

1. Delphi справочное руководство. Borland International, Inc. 1997.
2. InterBase справочное руководство. Borland International, Inc. 1996.
3. А.Жуков. Изучаем Delphi. Санкт-Петербург, Питер 2000.
4. Добудько Т.В., Пугач В.И. Открытые программные комплексы «Роботландия». Самара, 1996.
5. Добудько Т.В., Пугач В.И. Програмируем на Pascal. Самара, 1996.
6. Документация в электронном виде к Borland Pascal 7.0
7. Документация в электронном виде к CAD/CAM системе Adem for Windows 3.03.
8. Документация в электронном виде к Delphi 3.0
9. Документация в электронном виде к музыкальному конструктору Dance Machine.
10. Документация в электронном виде к музыкальному редактору «Шарманщик».
11. Документация в электронном виде к музыкальному редактору S cakewalk Professional.
12. Документация в электронном виде к музыкальному редактору Sound Forge.
13. Документация в электронном виде к языку программирования Basic.
14. Инженерная графика. Методические указания к лабораторным работам для студентов Тольяттинского университета им. Татищева. Тольятти, 1996.
15. Н.Вирт. Алгоритмы и структуры данных. Москва «Мир» 1989.
16. Н.Рубенкинг. Турбо Паскаль для Windows том 1. Москва «Мир», СК Ферлаг Интернешнл, 1994.
17. Рекламный буклет. Тольяттинский политехнический институт. Учебно-методический центр CAD/CAM.
18. Стефанюк В.Л. Компьютер обретает разум. Москва, «Мир», 1990.
19. Учебное пособие под редакцией Чемпинского Л.А. Компьютерные чертежно-графические системы для разработки конструкторской и технологической документации в машиностроении. Москва АСАДЕМА 2002.
20. Чемпинский Л.А. Выполнение чертежей на ПЭВМ. Методические указания к практическим занятиям. Самара 1994.
- Чемпинский Л.А. Инженерная графика на ПЭВМ. Самара, 19
21. Окулов С.М. Программирование в алгоритмах. М.: БИНОМ. Лаборатория знаний, 2002