

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Министерство образования Пермского края

Управление образования Чайковского городского округа

МБОУ Фокинская СОШ

УТВЕРЖДЕНО

Директор

МБОУ Фокинская СОШ

**приказ № 01-26-211 от «30»
августа 2023 г.**



РАБОЧАЯ ПРОГРАММА

учебного курса «Программирование»

для обучающихся 10-11 классов

Разработчик: Рыболова Л.А.

с. Фоки 2023

Пояснительная записка

Изучение основ программирования связано с целым рядом умений и навыков (организация деятельности, планирование и т.д.), которые по праву носят общеинтеллектуальный характер и формируются некоторыми – одна из приоритетных задач современной школы.

Очень велика роль изучения программирования для развития мышления школьников, формирования многих приёмов умственной деятельности.

Паскаль – идеальный язык для обучения программированию. Как ни странно, так оно и есть – в конце концов, он для того и создан. Сильных программистов в реальной жизни мало. Больше средних. Ещё больше слабых. И вот для большинства существует Паскаль. Он гарантирует получение приемлемого результата.

В экзаменационной работе Единого государственного экзамена (ЕГЭ) в части заданий требуются знания языка программирования. В формулировке задачи на выбор представлены следующие языки: Бейсик, Паскаль, Си, Алгоритмический язык, Python, Естественный

язык. Выбор языка не влияет на оценку, однако следует учитывать тот факт, что задания проверяют и оценивают (особенно это касается написания программ в заданиях ЕГЭ).

В рекомендациях ФИПИ проверяющим сказано: «Если этот язык программирования недостаточно знаком эксперту, то ему следует воспользоваться доступной справочной литературой или обратиться за помощью к консультанту или председателю (заместителю председателя) предметной комиссии, соблюдая при этом регламент проверки», т.е. проверяющий с неизвестным ему языком должен найти в очень сжатые сроки того, кто этот язык программирования знает и разберётся в тексте программы. Очевидно, что вероятность недооценки очень велика. Для естественного языка отсутствуют официальные правила, описывающие язык. Поэтому лучшими на ЕГЭ остаются: Паскаль, Си, Бейсик. А у языка Паскаль есть ещё один плюс: изначально сложная 27-я задача решалась именно на нём, а значит примеры, образцы решений и разборы типов задач, легко найти в литературе и в интернете, что нельзя сказать о других языках.

По озвученным выше причинам авторская программа элективного курса «Программирование» предусматривает обучение программированию на языке Паскаль в версии Pascal ABC и, соответственно, подготовку к ЕГЭ в части заданий, требующих знаний языка программирования.

Почему именно Pascal ABC? Следующая цитата от разработчиков Pascal ABC, многое объясняющая:

«Язык Паскаль признан многими российскими преподавателями как один из лучших именно для начального обучения. Однако, среда Borland Pascal, ориентированная на MS-DOS, устарела, а среда Borland Delphi её богатыми возможностями сложна для начинающего программиста. Так, попытка начинать обучение написанию событийной программы в Borland Delphi вызывает у обучаемого массу сложностей и приводит к ряду неправильно сформированных навыков».

Pascal ABC, как подчёркивают его создатели, это абсолютно учебный язык, программа медленно и торжественно разворачивается сверху вниз (традиционный язык – тот, который, что вижу – то пою). Очень сильно в Delphi не хватает этой маленькой радости – того, что у программы есть начало и есть конец. Поэтому начинать программировать сразу на Delphi (или аналогичном средстве разработки) – это беспощадная травма для начинающего программиста, он никогда не сможет стать сильным программистом, в лучшем случае – средним.

BorlandPascal, как понятно, синоним TurboPascal (TP). Отличается TurboPascal отPascalABCнебольшее, чемдесятьюпроцентами. Поэтомуэлективныйкурстакжепредусматривает в конце каждой темы дополнительное разъяснение, чем работа сTurboPascal отличается от Pascal ABC. При желании учащегосяпрограммировать на TP, он всёсможетсделатьсам.

Программасоставленанаоснове:

- книги Н.Ю. Комлев «Самоучитель игры на Паскале. ABC и немного Турбо» - М.:СОЛОН-ПРЕСС,2019.-256 с.;

- заданийЕГЭ

ирассчитананаучащихся10-хи11-

хклассов,изучающихпредметинформатикуиИКТпо1 часовойпрограммеавт.СемакинИ.Г.,Хеннер Е.К.,ШейнаТ.Ю(базовыйкурс).

Основныецелиизадачкурса:

- Формированиеинтересакизучениюпрофессии,связаннойспрограммированием.
- Формированиеалгоритмическойкультуры.
- Датьученикувозможностьреализоватьсвойинтересквыбранномукурсу.
- Научитьучащихсяструктурномупрограммированиюкакметоду,предусматривающемусозданиепонятных,локальнопростыхиудобночитаемыхпрограмм,характернымиособенностямикоторыхявляется:модульность,использованиеунифицированныхструктурследования,выбораиповторения,отказотнеструктурированных передач управления,ограниченноеиспользованиеглобальныхпеременных.
- Приобретениезнанийинавыковалгоритмизацииучащимисявееструктурномварианте.
- Освоениевсевозможныхметодоврешениязадач,реализуемыхнаязыкеПаскаль.
- Развитиеалгоритмическогомышления.
- Формированиенавыковграмотнойразработкипрограмм.
- ПомощьвподготовкекЕГЭпоинформатикевчастипрограммирования.
- Углублениезнаний,уменийинавыковрешениязадачпопрограммированиюиалгоритмизации.

1. Планируемыерезультаты

Врамкахданногокурсаучащиесяовладеваютследующимизнаниями,умениямииспособами деятельности:

- Знаютипонимаютрольпрограммнообеспеченияеговиды;
- Знают,чтотакоемногомерные(двумерные)массивы,умеютихобрабатывать;
- ЗнаютиреализуютпроцедурыифункциинаязыкеPascal;
- Умеютработатьсрекурсивнымиалгоритмами;
- Умеютчитатьтехническоезаданиеиписатьпонемупрограммынаязыке Pascal;
- УмеютотладитьнаписаннуюнаязыкеPascalпрограммувинтерактивномрежиме;

- Программируют на языке Pascal собственную разработанную задачу (алгоритм), созданную (созданный) по поставленному техническому заданию.

Предметом диагностики и контроля являются составленные алгоритмы и программы на языке Pascal предложенным задачам.

2. Содержание курса

Данный элективный курс представляет собой расширение и дополнение к названному базовому курсу информатики и информационным технологиям по разделу

«Программирование обработки информации» является для учащихся предметом выбора.

Вопросы, рассматриваемые в курсе, выходят за рамки обязательного содержания. Вместе с тем, они тесно примыкают к основному курсу. Поэтому данный элективный курс будет способствовать совершенствованию и развитию важнейших УУД в области информатики, предусмотренных школьной программой, поможет оценить свои возможности по информатике и более осознанно выбрать профиль дальнейшего обучения.

Курс рассматривает информатику как средство развития логического мышления, умение анализировать, выявлять сущности и отношения, описывать планы действий, делать логические выводы.

Содержание курса представляет самостоятельный модуль, который состоит из теоретических и практических занятий. Теоретическая часть проводится в урочной форме. Практическую часть занятия обучающиеся проводят в режиме консультаций с преподавателем, и после каждого занятия предполагается самостоятельная отработка обучающимися материала по каждой теме курса.

Согласно федеральному базисному учебному плану для образовательных учреждений Российской Федерации программа элективного курса рассчитана на 68 часов, по 1 часу в неделю (в 10-м классе – 34 часа и в 11-м классе – 34 часа).

Весь курс разбит на отдельные модули – разделы. *Текущий контроль* уровня усвоения материала осуществляется по результатам выполнения практических заданий, изучаемых в каждом разделе. *Итоговый контроль* – по результатам выполнения контрольного задания по окончанию изучения модуля и по прохождению всего курса, в виде *итогового тестирования*.

Тематическое планирование (10 класс)

№	Название темы	Количество часов		
		общее	теория	практика
1	<i>Введение (обобщение знаний по изученному материалу за предыдущий элективный курс 10 класса)..</i>	<i>1</i>	<i>1</i>	-
2	<i>Двумерные массивы</i>	<i>5</i>	<i>2</i>	<i>3</i>
3	<i>Процедуры и функции</i>	<i>6</i>	<i>2</i>	<i>4</i>
4	<i>Рекурсия</i>	<i>5</i>	<i>1</i>	<i>4</i>
5	<i>Отладка программ в интерактивном режиме</i>	<i>5</i>	<i>1,5</i>	<i>3,5</i>
6	<i>Написание программ по поставленным техническим заданиям</i>	<i>4</i>	<i>1</i>	<i>3</i>

7	<i>Работа с текстовыми файлами</i>	2	1	1
8	<i>Реализация алгоритма посредством написания задачи на языке Pascal по поставленной задаче</i>	3	1	2
9	<i>Повторение+итоговое тестирование+анализ</i>	3	1,5	1,5
Итого:		34	12	22

Тематическое планирование(11 класс)

№	Название темы	Количество часов		
		общее	теория	практика
1	<i>Введение (обобщение знаний по изученному материалу за предыдущий элективный курс 10 класса)..</i>	1	1	-
2	<i>Двумерные массивы</i>	5	2	3
3	<i>Процедуры и функции</i>	6	2	4
4	<i>Рекурсия</i>	5	1	4
5	<i>Отладка программ в интерактивном режиме</i>	5	1,5	3,5
6	<i>Написание программ по поставленным техническим заданиям</i>	4	1	3
7	<i>Работа с текстовыми файлами</i>	2	1	1
8	<i>Реализация алгоритма посредством написания задачи на языке Pascal по поставленной задаче</i>	3	1	2
9	<i>Повторение+итоговое тестирование+анализ</i>	3	1,5	1,5
Итого:		34	12	22

4. Календарно-

тематическое поурочно-планирование (10 класс) (Элективный курс «Программирование»)

№	Тема урока	Кол-во час.	Элементы содержания урока	Предметные результаты
1	Знакомство со средой программирования PascalABC.	1	Цели и задачи изучения курса в 10-11 классах, какие темы будут изучены. Общие сведения о языке Pascal, средах разработки, в частности, среде разработки PascalABC.	<u>Учащиеся должны знать:</u> – историю развития языка Pascal; – как компилятор среды для разработки на языке Pascal существуют – чем отличаются различные среды компиляторы языка Pascal;
2	Ввод, компиляция и запуск программы на языке Pascal.	1	Знакомство с графической средой PascalABC. Ввод простейшей программы. Компиляция, запуск и выполнение программы с использованием среды PascalABC.	<u>Учащиеся должны уметь:</u> – установить и запустить среду PascalABC; – вводить текст программы в текстовом редакторе среды PascalABC; – компилировать и запускать программу в среде PascalABC.
3	Простейший ввод и вывод данных на экран (без использования переменных).	1	Основные функции для ввода и вывода данных на экран: Read, ReadLn, Write, WriteLn.	<u>Учащиеся должны знать:</u> – конкретно назначение каждой функции ввода-вывода; – синтаксис вызова функций ввода-вывода.
4	Выполнение упражнений на ввод и вывод в PascalABC.	1	Ввод и вывод значений. Рисование псевдографических узоров на экране.	<u>Учащиеся должны знать:</u> – правила объявления переменных; <u>Учащиеся должны уметь:</u> – рисовать на экране псевдографические узоры при помощи Write и WriteLn.
5	Переменные. Объявление переменных. Присваивание переменных. Арифметические операции.	1	Правила объявления и именования переменных базовых типов Integer и Real. Присваивание значений переменным. Базовые арифметические операции: сложение, вычитание, умножение, деление.	<u>Учащиеся должны знать:</u> – правила объявления переменных и присваивания им значения; – базовые арифметические операции с численными типами данных.
6	Написание программ с простыми арифметическими действиями.	1	Решение типовых задач, выполняющих арифметические операции над базовыми типами Integer и Real.	<u>Учащиеся должны уметь:</u> – правильно применять арифметические операторы согласно поставленной задаче.
7	Простые типы данных (целочисленные, вещественные, строковые, символьные). Ввод и вывод типов данных.	1	Целочисленные типы данных. Вещественные типы данных. Строковые типы данных. Символьные типы данных. Булевы типы данных. Ввод и вывод переменных различного типа. Операции преобразования типов данных: round, trunc, chr, ord и др.	<u>Учащиеся должны знать:</u> – все простые типы данных, которые предоставляются языком Pascal; – характеристики различных типов данных и их отличия; – базовые арифметические операции с семитипами данных; – способы преобразования типов данных из одного вида в другой.
8	Практика написания программ, использующих различные типы данных.	1	Решение более сложных задач, выполняющих арифметические действия и преобразования различных типов данных.	<u>Учащиеся должны уметь:</u> – правильно применять арифметические операторы согласно поставленной задаче; – преобразовывать типы данных из одного формата в другой.

9	Алгоритм для конкретного исполнителя с фиксированным набором команд. Условные операторы if и else.	1	Ветвление алгоритмов. Операторы if и else. Блоки begin и end. Примеры задач.	<u>Учащиеся должны знать:</u> – синтаксис операторов if и else; – синтаксис блока команд begin/end.
10	Написание программ ветвления алгоритмов в припомощи операторов if и else.	1	Решение задач на написание алгоритмов ветвления.	<u>Учащиеся должны уметь:</u> – писать программы ветвления алгоритмических вычислений припомощи операторов if и else; – пользоваться составными операторами begin/end.
11	Усложнённые ветвления с использованием условных операторов if и else.	1	Составные вложенные операторы if и else. Примеры задач.	<u>Учащиеся должны знать:</u> – правила написания составных вложенных операторов ветвления; – принципы выделения основных веток прохождения алгоритма.
12	Практика написания программ усложнёнными ветвлениями.	1	Решение задач со сложными ветвлениями алгоритмов	<u>Учащиеся должны уметь:</u> – правильно выделять ветви прохождения алгоритма; – правильно пользоваться составными вложенными операторами if и else; – проверять алгоритм на прохождении всех веток.
13	Выполнение контрольного задания по пройденному материалу.	1	Решение комплексной задачи на ввод-вывод, арифметические операции, составные операторы и ветвления.	<u>Учащиеся должны уметь:</u> – самостоятельно решать сложные задачи применением полученных по темам 1-12 знаний и навыков.
14	Цикл for.	1	Понятие цикла и его назначение. Использование переменной в качестве счётчика итераций. Цикл for с увеличением и уменьшением счётчика.	<u>Учащиеся должны знать:</u> – принцип работы цикла for; – способы задания верхней и нижней границы счётчика; – различие между циклом for...to и for...downto.
15	Практика написания программ с циклом for.	1	Решение задач с использованием цикла for.	<u>Учащиеся должны знать:</u> – синтаксис и принцип работы цикла for. <u>Учащиеся должны уметь:</u> – решать задачи с использованием цикла for.
16	Циклы while и repeat until.		Цикл while. Цикл repeat...until. Операторы break и continue.	<u>Учащиеся должны знать:</u> – принцип работы циклов while и repeat...until; – различие между циклами while и repeat...until; – способ досрочного выхода из цикла и перехода к следующей итерации.
17	Практика написания программ с циклами while и repeat...until.	1	Решение задач с использованием циклов while и repeat...until.	<u>Учащиеся должны знать:</u> – синтаксис и принцип работы циклов while и repeat...until. <u>Учащиеся должны уметь:</u> – решать задачи с применением циклов while и repeat...until.
18	Анализ программ (разбор задач заданий ЕГЭ).	1	Разбор решений задач заданий для ЕГЭ	<u>Учащиеся должны уметь:</u> – решать задачи из задания №8 ЕГЭ.
19	Анализ программ с циклами и условными операторами (разбор задач заданий ЕГЭ).	1	Разбор решений задач заданий для ЕГЭ	<u>Учащиеся должны уметь:</u> – разбирать написанный код при применении циклов; – решать задачи из задания №20 ЕГЭ.
20	Выполнение контрольного задания по пройденному материалу.	1	Самостоятельное решение задач из сборника задач ЕГЭ.	<u>Учащиеся должны уметь:</u> – самостоятельно решать задачу №8 ЕГЭ; – самостоятельно решать задачу №20 ЕГЭ.

21	Работа со строковыми переменными.	1	Основные функции для работы со строковыми переменными: конкатенация, копирование, вставка, поиск, удаление,	<u>Учащиеся должны знать:</u> – что такое подстрока; – назначение функций Length, Copy, Pos, Val, Str, Concat, Insert, Delete, FillChar.
----	-----------------------------------	---	---	--

			доступ произвольному символу.	
22	Практика работы с трюковыми переменными.	1	Решение задач по работе с трюками.	<u>Учащиеся должны уметь:</u> – применять знания функций манипуляции трюками на практике; – решать задачи, осуществляющие преобразование текстовой информации.
23	Вспомогательные функции для преобразования данных. Математические функции.	1	Дополнительные математические функции, позволяющие организовывать вычисления с вещественными числами.	<u>Учащиеся должны знать:</u> – тригонометрические функции \arctan , \sin , \cos и их область определения; – степенные функции \exp , $\sqrt{\quad}$, $\sqrt{\quad}$ и их область определения; – функцию получения модуля числа – abs .
24	Одномерные массивы.	1	Определение массива, правила объявления одномерного массива, правила индексации элементов.	<u>Учащиеся должны знать:</u> – что такое массивы; <u>Учащиеся должны уметь:</u> – объявлять массивы в языке Pascal; – читать и писать элементы массива по заданному индексу.
25	Обработка одномерных массивов с использованием циклов. Задача поиска \min и \max элементов массива.	1	Итерация по массивам. Ввод данных в массив. Решение типовых задач на поиск по массивам.	<u>Учащиеся должны уметь:</u> – писать код, позволяющий вводить данные массива; – писать циклы, осуществляющие необходимый поиск данных в массиве.
26	Задачи вычисления значений на основе элементов массива.	1	Решение задач на различные вычисления, основанные на данных, хранящихся в массиве.	<u>Учащиеся должны уметь:</u> – на основании поставленного алгоритма написать программу, получающую на входе данные массива, и выводящую соответствующий результат; – уметь читать и понимать подобные программы, написанные другими людьми.
27	Задача сортировки массива.	1	Задача сортировки. Базовые алгоритмы сортировки: пузырьковая, выборочная, вставками, слияниями, быстрая сортировка.	<u>Учащиеся должны знать:</u> – задачи, решаемые алгоритмами сортировки; – положительные и отрицательные стороны каждого базового алгоритма сортировки.
28	Практика написания программы обработки данных, хранящихся в одномерном массиве.	1	Написание программ, выполняющих различные алгоритмы обработки данных в массивах.	<u>Учащиеся должны уметь:</u> – реализовывать поставленный в качестве задания алгоритм обработки одномерных массивов; – понимать алгоритм исходного кода программы.
29	Практика написания программы обработки данных, хранящихся в одномерном массиве.	1	Написание программ, выполняющих различные алгоритмы обработки данных в массивах.	<u>Учащиеся должны уметь:</u> – реализовывать поставленный в качестве задания алгоритм обработки одномерных массивов; – понимать алгоритм исходного кода программы.
30	Выполнение контрольного задания по теме "Обработка одномерных массивов с использованием циклов".	2	Решение контрольных заданий по обработке данных в массивах.	<u>Учащиеся должны уметь:</u> – реализовывать поставленный в качестве задания алгоритм обработки одномерных массивов; – понимать алгоритм исходного кода программы.
31	Разбор задач заданий ЕГЭ.	1	Решение примеров задачи ЕГЭ.	<u>Учащиеся должны уметь:</u> – самостоятельно писать программы, решающие поставленную задачу.
32	Практика по задачам	1	Решение примеров задачи ЕГЭ.	<u>Учащиеся должны уметь:</u> – самостоятельно писать программы, решающие поставленную задачу.
33	Выполнение контрольного задания	1	Контрольная работа с решением задач ЕГЭ.	<u>Учащиеся должны уметь:</u> – самостоятельно писать программы,

	по пройденному ма териалу. Тестирование по заданию №19 ЕГЭ.			решающие поставленную задачу.		
34	Повторение пройденного материала.	1	Обобщение всего пройденного ранее материала, освещение тем, которые будут проходить в следующем учебном году.	<u>Учащиеся должны знать:</u> – перечень пройденного материала по программированию; – перечень тем вопросов, которые будут ос вещены в следующем учебном году.	29.05.22	

Календарно-

тематическое поурочное планирование (1 класс) (Элективный курс «Программирование»)

№	Тема урока	Кол-во часов	Элементы содержания урока	Предметные результаты
1	Введение (обобщение знаний по изученному материалу)	1	Краткое обобщение знаний, пройденных в прошлом году. Объявление целей и задач, предстоящих в этом учебном году.	<u>Учащиеся должны знать:</u> – материал, пройденный в прошлом году; – перечень знаний, умений и навыков, которые они получат в этом учебном году.
2	Двумерные массивы.	1	Определение двумерного массива, способы его задания, правила индексирования элементов.	<u>Учащиеся должны уметь:</u> – объявлять двумерные массивы; – индексировать конкретные элементы двумерного массива.
3	Обработка многомерных (двумерных) массивов с использованием циклов. Вычисление данных массива.	1	Типовые алгоритмы работы с двумерными массивами: ввод данных, поиск, вычисление.	<u>Учащиеся должны знать:</u> – правила ввода двумерных массивов; – типовые алгоритмы поиска под двумерным массивом; – принципы вычисления данных на основе двумерных массивов.
4	Обработка многомерных (двумерных) массивов с использованием циклов. Инвертирование элементов массива.	1	Понятие многомерного массива. Типовые алгоритмы трансформации массивов.	<u>Учащиеся должны знать:</u> – правила объявления многомерных массивов; <u>Учащиеся должны уметь:</u> – трансформировать содержимое многомерных массивов по заданному алгоритму.
5	Практика работы с многомерными массивами.	1	Решение задач на многомерные массивы.	<u>Учащиеся должны уметь:</u> – решать задачи на ввод данных, поиск, вычисления и трансформации данных с использованием многомерных массивов.
6	Выполнение контрольного задания по пройденной теме "Двумерные массивы".	1	Самостоятельное решение задач по материалам, пройденным на занятиях 2-4.	<u>Учащиеся должны уметь:</u> – самостоятельно решать поставленные задачи по работе с многомерными массивами.
7	Процедуры и функции.	1	Понятие процедуры. Понятие функции. Объявление процедур и функций. Правила передачи параметров в процедуры и функции. Возвращение результата функцией. Вложенные процедуры и функции. Досрочный выход из процедур и функций.	<u>Учащиеся должны знать:</u> – суть процедуры и функций и их назначение; – правила передачи аргументов по значению и по ссылке; – синтаксис возврата значения из функций; – досрочный выход из подпрограммы с помощью инструкции <code>exit</code> .
8	Практика написания программ с использованием процедур и функций.	1	Решение задач с использованием процедур и функций.	<u>Учащиеся должны уметь:</u> – определять дублирующиеся участки алгоритмов и выносить их в подпрограммы; – правильно определять перечень дополнительных аргументов, передаваемых в подпрограммы.

9	Практика написания программы использованием	1	Решение задач с использованием процедур и функций.	<u>Учащиеся должны уметь:</u> – писать программы, содержащие несколько
---	---	---	--	---

	процедуры функций.			процедур/функций одновременно.
10	Анализ программ с циклами подпрограммами. Разбор задач ЕГЭ.	1	Решение задачи №21 из примеров ЕГЭ по информатике.	<u>Учащиеся должны уметь:</u> – понимать суть поставленной задачи; – решать задачу из ЕГЭ.
11	Анализ программ с циклами подпрограммами. Разбор задач заданий ЕГЭ.	1	Решение задачи №21 из примеров ЕГЭ по информатике.	<u>Учащиеся должны уметь:</u> – решать задачу из примеров ЕГЭ; – понимать код, представленный в тексте задачи.
12	Выполнение контрольного задания по пройденной теме "Процедуры и функции".	1	Решение задач, подразумевающих использование процедур функций, а также примеров задачи №21 из ЕГЭ.	<u>Учащиеся должны уметь:</u> – самостоятельно писать программы, содержащие несколько процедур/функций одновременно; – самостоятельно решать задачу из примеров ЕГЭ.
13	Рекурсия. Разбор рекурсивных алгоритмов.	1	Понятие рекурсии. Примеры простых рекурсивных алгоритмов: факториал, расчёт числа Фибоначчи.	<u>Учащиеся должны знать:</u> – определение рекурсивного алгоритма; – отличие алгоритма с явной рекурсией от алгоритма с косвенной рекурсией; – преимущества и недостатки рекурсивного решения задачи по отношению к итеративному.
14	Практика решения задачи на рекурсию.	1	Решение задач, содержащих явно или косвенно определённый рекурсивный алгоритм.	<u>Учащиеся должны уметь:</u> – реализовывать программы с использованием рекурсивных алгоритмов.
15	Практика решения задачи на рекурсию. Решение задач заданий ЕГЭ.	1	Решение задачи №1 из примеров ЕГЭ с использованием рекурсии.	<u>Учащиеся должны знать:</u> – принципы построения рекурсивных алгоритмов; – способы решения задачи ЕГЭ.
16	Практика решения задачи на рекурсию. Решение задач заданий ЕГЭ.	1	Решение задачи №1 из примеров ЕГЭ с использованием рекурсии.	<u>Учащиеся должны уметь:</u> – разбирать и понимать предоставленные исходные тексты программ, содержащих рекурсию.
17	Выполнение контрольного задания по теме "Рекурсия".	1	Самостоятельное решение задач по пройденным темам 13-16.	<u>Учащиеся должны уметь:</u> – самостоятельно формулировать и реализовывать в виде программы рекурсивный алгоритм; – самостоятельно решать задачу ЕГЭ.
18	Отладка программ, написанных на языке Pascal.	1	Проблема анализа и решения сложных задач. Средства и механизмы, позволяющие отлаживать программы. Запуск и отладка сложных программ.	<u>Учащиеся должны знать:</u> – механизмы, позволяющие отлаживать программы; – средства, предоставляемые средой PascalABC для отладки программ; – принципы отладки программ.
19	Практика отладки программ в интерактивном режиме.	1	Практика отладки нетривиальных программ с помощью отладчика.	<u>Учащиеся должны знать:</u> – каким образом отлаживается программа; – назначение точек останова; – назначение точек. <u>Учащиеся должны уметь:</u> – запускать программу в отладчике; – пользоваться точками

				<p>останова;</p> <ul style="list-style-type: none"> – работать в пошаговом режиме выполнения программы; – читать и определять значения переменных.
20	Отладка программиззаданий ЕГЭ (поисклогическихшибок).		Решение задачи № 24 ЕГЭ на поисклогическихшибок. Демонстрацияэффективностинахожденияэтихлогическихшибок с помощьюотладчика.	<p><u>Учащиесядолжнызнать:</u></p> <ul style="list-style-type: none"> – как осуществляется поисклогическихшибок в программебезотладчика; – как осуществляется поисклогическихшибок в программесотладчиком.
21	Отладкапрограммиззаданий ЕГЭ (поисклогическихшибок).	1	Решение задачи № 24 ЕГЭ сиспользованиемотладчикаибезотладчика.	<p><u>Учащиесядолжныуметь:</u></p> <ul style="list-style-type: none"> – находить логическиеошибки в неверномрешении поставленнойзадачи.
22	Выполнениепрактическогоконтрольного заданияпо отладке программ винтерактивномрежиме . Тестированиепо заданиюЕГЭ.	1	Самостоятельноерешениезадачнапоиск и устранение логическихшибокв программе.	<p><u>Учащиесядолжныуметь:</u></p> <ul style="list-style-type: none"> – самостоятельнорешатьиустранять логическиеошибки в программе; – самостоятельнорешатьзадачуЕГЭ.
23	Написание программ попоставленнымтехническимзаданиям.	1	Понятие технического задания.Требования к программе. Стадиипроектирования и реализации.Оценкавременисебестоимости.	<p><u>Учащиесядолжнызнать:</u></p> <ul style="list-style-type: none"> – что являетсятехническимзаданием; – какиетребованияк программам могутвыдвигаться; – какие стадии развитияпроходит любоепрограммноеобеспечение; – какоцениваютсясрокии себестоимость выполненияпроекта.
24	Практика написанияпрограмм попоставленнымтехническимзаданиям.	1	Написание программ попоставленному техническому заданию.	<p><u>Учащиесядолжныуметь:</u></p> <ul style="list-style-type: none"> – анализироватьтехническоезадание; – проектироватьпрограммное обеспечениесогласнопоставленному техническому заданию.
25	Практика написанияпрограмм попоставленнымтехническим заданиям.Работа с заданиемиз ЕГЭ.	1	Решениепримеровзадачи№25изпримеровЕГЭ.	<p><u>Учащиесядолжныуметь:</u></p> <ul style="list-style-type: none"> – решатьзадачуизЕГЭ.
26	Написаниеконтрольного заданияпо заданномутехническому заданию. Тестирование позаданию№25 ЕГЭ.	1	Самостоятельноерешениезадачпоставленномутехническому заданию,атакжезадачи№25изЕГЭ.	<p><u>Учащиесядолжныуметь:</u></p> <ul style="list-style-type: none"> – самостоятельноанализироватьпоставленное техническоезадание; – самостоятельнонаходитьпути по решениюпоставленнойзадачи; – самостоятельнорешатьзадачу№25ЕГЭ.
27	Работа с текстовымифайлами (загрузка исохранениеданныхиз файла).	1	Понятие файла. Специальныефайловыетипыданных. Функциидляработысфайлами.	<p><u>Учащиесядолжнызнать:</u></p> <ul style="list-style-type: none"> – определениефайловыхтипов данных; – основныефункциидляработысфайлами: Assign,Reset,Rewrite,Read,Write,ReadLn, WriteLn,BlockRead,Block Write, Seek,Close,Eraseidr.

28	Практика работы с текстами.	1	Чтение и обработка текстовых файлов.	<u>Учащиеся должны уметь:</u> – писать программы, осуществляющие файловый ввод-вывод.
29	Реализация алгоритма посредством написания задачи на языке Pascal по поставленной задаче.	1	Решение более сложных задач повсемуранее пройденному материалу.	<u>Учащиеся должны уметь:</u> – понимать поставленную задачу; – применять все ранее полученные знания; – уметь находить корректное решение поставленной задачи.
30	Реализация алгоритма посредством написания задачи на языке Pascal по поставленной задаче.	1	Решение более сложных задач повсемуранее пройденному материалу.	<u>Учащиеся должны уметь:</u> – понимать поставленную задачу; – применять все ранее полученные знания; – уметь находить корректное решение поставленной задачи.
31	Реализация алгоритма посредством написания задачи на языке Pascal по поставленной задаче.	1	Решение более сложных задач повсемуранее пройденному материалу.	<u>Учащиеся должны уметь:</u> – понимать поставленную задачу; – применять все ранее полученные знания; – уметь находить корректное решение поставленной задачи.
32	Повторение пройденного материала за курс.	1	Подытоживание всего пройденного материала.	<u>Учащиеся должны знать:</u> – весь материал, пройденный за курс 10-11 классов по программированию на языке Pascal. <u>Учащиеся должны уметь:</u> – самостоятельно находить решения поставленным задачам в рамках пройденного курса.
33	Итоговое тестирование.	1	Самостоятельное решение задач повышенной сложности на языке Pascal.	<u>Учащиеся должны уметь:</u> – самостоятельно находить решения к поставленным задачам в рамках пройденного курса.
34	Анализ результатов.	2	Анализ результатов итогового тестирования. Разбор задач.	<u>Учащиеся должны знать:</u> – весь материал, пройденный за курс 10-11 классов по программированию на языке Pascal.