

Муниципальное общеобразовательное учреждение

«Средняя школа № 2»

Рассмотрена на заседании ПС

протокол № 12 от 27.08.2020 г

Рассмотрена на заседании УС школы

Протокол №3 от 28.08.2020 г

Утверждена приказом муниципального

общеобразовательного учреждения

«Средняя школа № 2»

01-02/266 от 27.08.2020 г

Директор школы:

А.Л.Розина



Рабочая программа

на 2020-2021 учебный год

по предмету информатика

8 класс

Учитель: Кузьмицкая Л.Р., Карева И.Л.

Ярославль 2020 г.

1. Личностные, метапредметные и предметные результаты

Основными личностными результатами, формируемыми при изучении информатики в 8 классе являются:

- наличие представлений об информации как важнейшем стратегическом ресурсе развития личности, государства, общества;
- развитие чувства личной ответственности за качество окружающей информационной среды;
- способность увязать учебное содержание с собственным жизненным опытом, понять значимость подготовки в области информатики и ИКТ в условиях развития информационного общества;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Основными метапредметными результатами, формируемыми при изучении информатики в 8 классе являются:

- владение общепредметными понятиями «алгоритм», «исполнитель»;
- владение информационно-логическими умениями: определять понятия, классифицировать, самостоятельно выбирать основания и критерии для классификации;
- владение умениями самостоятельно планировать пути достижения целей; соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией; оценивать правильность выполнения учебной задачи;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий;
- владение информационным моделированием как основным методом приобретения знаний: умение выбирать форму представления информации в зависимости от стоящей задачи.

<i>Содержание</i>	<i>Предметные результаты</i>
Математические основы информатики	Научится: <ul style="list-style-type: none">• записывать в двоичной системе целые числа от 0 до 1024;• переводить заданное натуральное число из десятичной записи в двоичную и из двоичной в десятичную;• сравнивать числа в двоичной записи;• складывать и вычитать числа, записанные в двоичной системе счисления;• составлять и записывать логические выражения с операциями И, ИЛИ, НЕ и скобок;• определять истинность такого составного высказывания, если известны значения истинности входящих в него элементарных высказываний;• определять количество элементов в множествах, полученных из двух или трех базовых множеств с помощью операций объединения, пересечения и дополнения.

	<p><i>Выпускник получит возможность:</i></p> <ul style="list-style-type: none"> • научиться решать логические задачи с использованием таблиц истинности; • научиться решать логические задачи путем составления логических выражений и их преобразования с использованием основных свойств логических операций; • ознакомиться с влиянием ошибок измерений и вычислений на выполнение алгоритмов управления реальными объектами (на примере учебных автономных роботов).
Основы алгоритмизации	<p>Научится:</p> <ul style="list-style-type: none"> • понимать смысл понятия «алгоритм» и широту сферы его применения; анализировать предлагаемые последовательности команд на предмет наличия у них таких свойств алгоритма как дискретность, детерминированность, понятность, результативность, массовость; • оперировать алгоритмическими конструкциями «следование», «ветвление», «цикл» (подбирать алгоритмическую конструкцию, соответствующую той или иной ситуации); • выражать алгоритм решения задачи различными способами (словесным, графическим, в том числе и в виде блок-схемы, с помощью формальных языков и др.); • понимать термины «исполнитель», «формальный исполнитель», «среда исполнителя», «система команд исполнителя», «программа» и др., а также понимать разницу между употреблением этих терминов в обыденной речи и в информатике; понимать ограничения, накладываемые средой исполнителя и системой команд, на круг задач, решаемых исполнителем; • исполнять линейный алгоритм для формального исполнителя с заданной системой команд; • ученик научится исполнять записанный на естественном языке алгоритм, обрабатывающий цепочки символов; • составлять алгоритмы для решения учебных задач различных типов; • определять наиболее оптимальный способ выражения алгоритма для решения конкретных задач (словесный, графический, с помощью формальных языков); • выполнять без использования компьютера («вручную») несложные алгоритмы управления исполнителями; • составлять несложные алгоритмы управления исполнителями • определять результат выполнения заданного алгоритма или его фрагмента;

	<p><i>Выпускник получит возможность научиться:</i></p> <ul style="list-style-type: none"> • познакомиться с понятием «управление», с примерами того, как компьютер управляет различными системами (роботы, летательные и космические аппараты, станки, оросительные системы, движущиеся модели и др.); • познакомиться с учебной средой составления программ управления автономными роботами и разобрать примеры алгоритмов управления, разработанными в этой среде; • исполнять алгоритмы, содержащие ветвления и повторения, для формального исполнителя с заданной системой команд; • составлять все возможные алгоритмы фиксированной длины для формального исполнителя с заданной системой команд; • определять количество линейных алгоритмов, обеспечивающих решение поставленной задачи, которые могут быть составлены для формального исполнителя с заданной системой команд; • подсчитывать количество тех или иных символов в цепочке символов, являющейся результатом работы алгоритма.
Начала программирования	<p>Научится:</p> <ul style="list-style-type: none"> • составлять линейные алгоритмы, число команд в которых не превышает заданное; • исполнять линейные алгоритмы, записанные на алгоритмическом языке; • исполнять алгоритмы с ветвлениями, записанные на алгоритмическом языке; • понимать правила записи и выполнения алгоритмов, содержащих цикл с параметром или цикл с условием продолжения работы; • анализировать числовые и текстовые данные, записанные на конкретном языке программирования с использованием основных управляющих конструкций последовательного программирования (линейная программа, ветвление, повторение, вспомогательные алгоритмы); • анализировать числовые и текстовые данные с использованием основных управляющих конструкций последовательного программирования и записывать их в виде программ на выбранном языке программирования; выполнять эти программы на компьютере; • анализировать предложенный алгоритм, например, определять какие результаты возможны при заданном множестве исходных значений;

	<ul style="list-style-type: none"> • использовать логические значения, операции и выражения с ними; • записывать на выбранном языке программирования арифметические и логические выражения и вычислять их значения. <p><i>Выпускник получит возможность научиться:</i></p> <ul style="list-style-type: none"> • познакомиться с использованием в программах строковых величин и с операциями со строковыми величинами; • создавать программы для решения задач, возникающих в процессе учебы и вне ее; • познакомиться с задачами обработки данных и алгоритмами их решения; • по данному алгоритму определять, для решения какой задачи он предназначен.
--	---

2. Содержание учебного предмета

№	Тема	Ко-во часов	Содержание	Формы организации учебной деятельности и виды деятельности обучающегося
1.	Введение	1	Техника безопасности и правила работы на компьютере.	Форма организации работы учащихся: коллективная, групповая, индивидуальная. Форма организации обучения: урок
2.	Математические основы информатики	13	<p>Позиционные и непозиционные системы счисления. Примеры представления чисел в позиционных системах счисления. Основание системы счисления. Алфавит (множество цифр) системы счисления. Количество цифр, используемых в системе счисления с заданным основанием. Краткая и развернутая формы записи чисел в позиционных системах счисления.</p> <p>Двоичная система счисления, запись целых чисел в пределах от 0 до 1024. Перевод натуральных чисел из десятичной системы счисления в двоичную и из двоичной в десятичную.</p> <p>Восьмеричная и шестнадцатеричная системы счисления. Перевод натуральных чисел из десятичной системы счисления в восьмеричную, шестнадцатеричную и обратно.</p> <p>Перевод натуральных чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную и обратно.</p> <p><i>Арифметические действия в системах счисления.</i></p> <p><i>Представление целых и вещественных чисел в памяти компьютера</i></p> <p>Высказывания. Простые и сложные высказывания. Диаграммы Эйлера-Венна. Логические значения высказываний. Логические выражения. Логические операции: «и» (конъюнкция, логическое умножение), «или» (дизъюнкция, логическое сложение), «не» (логическое отрицание). Правила записи логических выражений. Приоритеты логических операций.</p>	<p>Форма организации работы учащихся: коллективная, групповая, индивидуальная.</p> <p>Форма организации обучения: урок</p> <p><i>Аналитическая деятельность:</i></p> <ul style="list-style-type: none"> • выявлять различие в унарных, позиционных и непозиционных системах счисления; • выявлять общее и отличия в разных позиционных системах счисления; • анализировать логическую структуру высказываний. <p><i>Практическая деятельность:</i></p> <ul style="list-style-type: none"> • переводить небольшие (от 0 до 1024) целые числа из десятичной системы счисления в двоичную (восьмеричную, шестнадцатеричную) и обратно; • выполнять операции сложения и умножения над небольшими двоичными числами; • строить таблицы истинности для логических выражений; • вычислять истинностное значение логического выражения.

№	Тема	Ко-во часов	Содержание	Формы организации учебной деятельности и виды деятельности обучающегося
			<p>Таблицы истинности. Построение таблиц истинности для логических выражений. Множество. Определение количества элементов во множествах, полученных из двух или трех базовых множеств с помощью операций объединения, пересечения и дополнения. <i>Логические операции следования (импликация) и равносильности (эквивалентность).</i></p> <p><i>Свойства логических операций. Законы алгебры логики.</i></p> <p>Расчет количества вариантов: формулы перемножения и сложения количества вариантов. Количество текстов данной длины в данном алфавите.</p> <p><i>Использование таблиц истинности для доказательства законов алгебры логики.</i></p> <p>Логические элементы. Схемы логических элементов и их физическая (электронная) реализация. Знакомство с логическими основами компьютера.</p>	
3.	Основы алгоритмизации	10	<p>Исполнители. Состояния, возможные обстановки и система команд исполнителя. Необходимость формального описания исполнителя. Ручное управление. Управление. Сигнал. Обратная связь. Примеры: компьютер и управляемый им исполнитель (в том числе робот); компьютер, получающий сигналы от цифровых датчиков в ходе наблюдений и экспериментов, и управляющий реальными (в том числе движущимися) устройствами. Алгоритм как план управления исполнителем (исполнителями). Алгоритмический язык (язык программирования) – формальный язык для записи алгоритмов. Понятие алгоритма как формального описания последовательности действий исполнителя при заданных начальных данных. Свойства алгоритмов. Составление алгоритмов</p>	<p>Форма организации работы учащихся: коллективная, групповая, индивидуальная.</p> <p>Форма организации обучения: урок</p> <p><i>Аналитическая деятельность:</i></p> <ul style="list-style-type: none"> • определять по блок-схеме, для решения какой задачи предназначен данный алгоритм; • анализировать изменение значений величин при пошаговом выполнении алгоритма; • определять по выбранному методу решения задачи, какие алгоритмические конструкции могут войти в алгоритм; • сравнивать различные алгоритмы решения одной задачи.

№	Тема	Ко-во часов	Содержание	Формы организации учебной деятельности и виды деятельности обучающегося
			<p>и программ по управлению исполнителями Робот, Черепашка, Чертежник и др.</p> <p>Программа – запись алгоритма на конкретном алгоритмическом языке. Компьютер – автоматическое устройство, способное управлять по заранее составленной программе исполнителями, выполняющими команды. Программное управление исполнителем. Программное управление самодвижущимся роботом. Словесное описание алгоритмов. Описание алгоритма с помощью блок-схем. Отличие словесного описания алгоритма, от описания на формальном алгоритмическом языке.</p> <p>Константы и переменные. Переменная: имя и значение. Типы переменных: целые, вещественные. Команда присваивания. Выражения.</p> <p>Конструкция «следование». Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных.</p> <p>Конструкция «ветвление». Полная форма ветвления Конструкция «ветвление». Неполная форма ветвления Конструкция «повторения»: с условием выполнения. Цикл с заданным условием продолжения работы.</p> <p>Конструкция «повторения»: с переменной цикла. Цикл с заданным условием окончания работы.</p> <p>Конструкция «повторения»: циклы с заданным числом повторений. Цикл с заданным числом повторений.</p>	<p><i>Практическая деятельность:</i></p> <ul style="list-style-type: none"> • исполнять готовые алгоритмы для конкретных исходных данных; • преобразовывать запись алгоритма с одной формы в другую; • строить цепочки команд, дающих нужный результат при конкретных исходных данных для исполнителя арифметических действий; • строить цепочки команд, дающих нужный результат при конкретных исходных данных для исполнителя, преобразующего строки символов; • строить арифметические, строковые, логические выражения и вычислять их значения
4.	Начала программирования	12	<p>Общие сведения о языке программирования Питон. Системы программирования. Типы переменных: целые,</p>	<p>Форма организации работы учащихся: коллективная, групповая, индивидуальная.</p>

№	Тема	Ко-во часов	Содержание	Формы организации учебной деятельности и виды деятельности обучающегося
	я		<p>вещественные, <i>символьные, строковые, логические</i>. Оператор присваивания. <i>Представление о структурах данных.</i></p> <p>Организация ввода и вывода данных. Средства создания и выполнения программ.</p> <p>Программирование линейных алгоритмов. Определение возможных результатов работы алгоритма при данном множестве входных данных; определение возможных входных данных, приводящих к данному результату. Примеры описания объектов и процессов с помощью набора числовых характеристик, а также зависимостей между этими характеристиками, выражаемыми с помощью формул.</p> <p>Условный оператор: полная и неполная формы. Выполнение и невыполнение условия (истинность и ложность высказывания). Простые и составные условия. Запись составных условий. Пример задачи обработки данных по нахождению минимального и максимального числа из двух, трех данных чисел. Знакомство с алгоритмами решением этой задачи. Реализации этих алгоритмов в выбранной среде программирования.</p> <p>Пример задачи обработки данных по нахождению всех корней заданного квадратного уравнения. Знакомство с алгоритмами решением этой задачи. Реализации этих алгоритмов в выбранной среде программирования. <i>Примеры записи команд ветвления в различных алгоритмических языках.</i></p> <p>Программирование циклов с заданным условием продолжения работы. <i>Понятие об этапах разработки программ и приемах отладки программ.</i></p> <p>Программирование циклов с заданным условием окончания</p>	<p>Форма организации обучения: урок</p> <p><i>Аналитическая деятельность:</i></p> <ul style="list-style-type: none"> • анализировать готовые программы; • определять по программе, для решения какой задачи она предназначена; • выделять этапы решения задачи на компьютере. <p><i>Практическая деятельность:</i></p> <ul style="list-style-type: none"> • программировать линейные алгоритмы, предполагающие вычисление арифметических, строковых и логических выражений; • разрабатывать программы, содержащие оператор/операторы ветвления (решение линейного неравенства, решение квадратного уравнения и пр.), в том числе с использованием логических операций; • разрабатывать программы, содержащие оператор (операторы) цикла

№	Тема	Ко-во часов	Содержание	Формы организации учебной деятельности и виды деятельности обучающегося
			<p>работы.</p> <p>Программирование циклов с заданным числом повторений.</p> <p>Запись алгоритмических конструкций в выбранном языке программирования. <i>Примеры записи команд повторения и других конструкций в различных алгоритмических языках. Проверка условия выполнения цикла до начала выполнения тела цикла и после выполнения тела цикла: постусловие и предусловие цикла. Инвариант цикла.</i> Понятие об этапах разработки программ: составление требований к программе, выбор алгоритма и его реализация в виде программы на выбранном алгоритмическом языке, отладка программы с помощью выбранной системы программирования, тестирование.</p>	

3. Календарно – тематическое планирование

№ урока	№ п/п	Тема урока	Содержание	8а	8б	8в	8г
1.	1.1	Цели изучения курса информатики и ИКТ. Правила техники безопасности на уроках информатики.	Цели изучения курса информатики и ИКТ. Правила техники безопасности на уроках информатики. Организация рабочего места.	1 четверть 10.09.20	07.09.20	03.09.20	14.09.20
Математические основы информатики (13 часов)							
2.	2.1	Общие сведения о системах счисления	Позиционные и непозиционные системы счисления. Примеры представления чисел в позиционных системах счисления. Основание системы счисления. Алфавит (множество цифр) системы счисления. Количество цифр, используемых в системе счисления с заданным основанием. Краткая и развернутая формы записи чисел в позиционных системах счисления.	10.09.20	07.09.20	03.09.20	14.09.20
3.	2.2	Двоичная система счисления.	Двоичная система счисления. Запись целых чисел в пределах от 0 до 1024. Перевод натуральных чисел из десятичной системы счисления в двоичную и из двоичной в десятичную.	24.09.20	21.09.2020	17.09.20	28.09.2020
4.	2.3	Восьмеричная и шестнадцатеричная системы счисления. Компьютерные системы счисления	Восьмеричная и шестнадцатеричные системы счисления. Перевод натуральных чисел из десятичной системы счисления в восьмеричную, шестнадцатеричную и обратно. Перевод натуральных чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную и обратно.	24.09.20	21.09.2020	17.09.20	28.09.2020
5.	2.4	Урок цифры «Искусственный интеллект и машинное обучение»	Урок цифры «Искусственный интеллект и машинное обучение»	08.10.20	05.10.2020	01.10.20	12.10.2020
6.	2.5	Двоичная арифметика	Двоичная арифметика.	08.10.20	05.10.2020	01.10.20	12.10.2020
7.	2.6	Двоичная арифметика Проверочная работа по теме «Системы счисления»	Двоичная арифметика.	22.10.20	19.10.2020	15.10.20	26.10.2020
8.	2.7	Высказывание. Логические операции.	Высказывания. Простые и сложные высказывания. Диаграммы Эйлера-Венна. Логические значения высказываний. Логические выражения. Логические операции: «и» (конъюнкция, логическое умножение), «или» (дизъюнкция, логическое сложение), «не» (логическое отрицание). Правила записи логических выражений. Приоритеты логических операций.	22.10.20	19.10.2020	15.10.20	26.10.2020
9.	2.8	Построение таблиц истинности для логических выражений.	Таблицы истинности. Построение таблиц истинности для логических выражений. Множество. Определение количества элементов во множествах, полученных из двух или трех базовых множеств с помощью операций объединения, пересечения и дополнения. <i>Логические операции следования (импликация) и равносильности (эквивалентность).</i>	2 четверть 19.11.20	2 четверть 16.11.20	29.10.20	23.11.20
10.	2.9	Свойства логических	<i>Свойства логических операций. Законы алгебры логики.</i>	19.11.20	16.11.20	29.10.20	23.11.20

		операций. Законы алгебры логики.	<i>Использование таблиц истинности для доказательства законов алгебры логики.</i>				
11.	2.10	<i>Всероссийский урок по безопасной работе в сети Интернет</i>		3.12.20	30.11.20	2 четверть 26.11.20	07.12.20
12.	2.11	Решение логических задач.	Решение логических задач. Расчет количества вариантов: формулы перемножения и сложения количества вариантов.	3.12.20	30.11.20	26.11.20	07.12.20
13.	2.12	Логические элементы	Логические элементы. Схемы логических элементов и их физическая (электронная) реализация. Знакомство с логическими основами компьютера. Знакомство с основными направлениями профессией: электронщик, инженер-конструктор радиоаппаратуры.	17.12.20	14.12.20	10.12.20	21.12.20
14.	2.13	Контрольная работа № 1 «Математические основы информатики»	Контрольная работа № 1 «Математические основы информатики»	17.12.20	14.12.20	10.12.20	21.12.20
2. Основы алгоритмизации (8 часов)							
15.	3.1	Алгоритмы и исполнители. Способы записи алгоритмов	Исполнители. Состояния, возможные обстановки и система команд исполнителя; команды-приказы и команды-запросы; отказ исполнителя. Необходимость формального описания исполнителя. Ручное управление исполнителем. Алгоритм как план управления исполнителем (исполнителями). Алгоритмический язык (язык программирования) – формальный язык для записи алгоритмов. Программа – запись алгоритма на конкретном алгоритмическом языке. Компьютер – автоматическое устройство, способное управлять по заранее составленной программе исполнителями, выполняющими команды. Программное управление исполнителем. <i>Программное управление самодвижущимся роботом.</i> Словесное описание алгоритмов. Описание алгоритма с помощью блок-схем. Отличие словесного описания алгоритма, от описания на формальном алгоритмическом языке.	3 четверть 14.01.21	3 четверть 11.01.21	24.12.20	3 четверть 18.01.21
16.	3.2	Объекты алгоритмов	Константы и переменные. Переменная: имя и значение. Типы переменных: целые, вещественные. Команда присваивания. Выражения. Составление алгоритмов и программ по управлению исполнителями Робот, Черепашка, Чертежник и др.	14.01.21	11.01.21	24.12.20	18.01.21
17.	3.3	Алгоритмическая конструкция следование. Исполнитель Робот.	Конструкция «следование». Линейный алгоритм. Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных. Управление. Сигнал. Обратная связь. Примеры: компьютер и управляемый им исполнитель (в том числе робот); компьютер, получающий сигналы от цифровых датчиков в ходе наблюдений и экспериментов, и управляющий реальными (в том числе движущимися) устройствами.	28.01.21	25.01.21	3 четверть 21.01.21	01.02.21
18.	3.4	Алгоритмическая конструкция «повторение».	Конструкция «повторения»: циклы с заданным числом повторений.	28.01.21	25.01.21	21.01.21	01.02.21

		Цикл с заданным числом повторений.					
19.	3.5	Алгоритмическая конструкция повторение. Цикл с заданным условием выполнения	Конструкция «повторения»: циклы с условием выполнения, с переменной цикла. <i>Проверка условия выполнения цикла до начала выполнения тела цикла и после выполнения тела цикла: постусловие и предусловие цикла. Инвариант цикла.</i>	11.02.21	08.02.21	04.02.21	15.02.21
20.	3.6	Алгоритмическая конструкция повторение. Цикл с переменной цикла	Конструкция «повторения»: циклы с переменной цикла. <i>Инвариант цикла. Примеры записи команд ветвления и повторения и других конструкций в различных алгоритмических языках.</i>	11.02.21	08.02.21	04.02.21	15.02.21
21.	3.7	Алгоритмическая конструкция ветвление. Условный оператор: полная и неполная формы.	Конструкция «ветвление». Условный оператор: полная и неполная формы. Выполнение и невыполнения условия (истинность и ложность высказывания). Простые и составные условия. Запись составных условий.	25.02.21	22.02.21 20.02.21	18.02.21	01.03.21
22.	3.8	Контрольная работа № 2 «Основы алгоритмизации»	Контрольная работа № 2 «Основы алгоритмизации»	25.02.21	22.02.21 20.02.21	18.02.21	01.03.21
Начала программирования (11 часов)							
23.	4.1	Системы программирования. Средства создания и выполнения программ.	Системы программирования. Средства создания и выполнения программ. Знакомство с языком программирования. Знакомство с направлениями работы в деятельности программиста.	11.03.21	08.03.21 22.03.21	04.03.21	15.03.21
24.	4.2	Организация ввода и вывода данных	Типы переменных: целые, вещественные, <i>символьные, строковые, логические</i> . Оператор присваивания. Организация ввода и вывода данных.	11.03.21	08.03.21 22.03.21	04.03.21	15.03.21
25.	4.3	Этапы разработки программы. Программирование линейных алгоритмов.	<i>Понятие об этапах разработки программ.</i> Простейшие приемы диалоговой отладки программ (выбор точки останова, пошаговое выполнение, просмотр значений величин, отладочный вывод).	25.03.21	4 четверть 12.04.21	18.03.21	4 четверть 05.04.21
26.	4.4	Программирование линейных алгоритмов	Программирование линейных алгоритмов. Определение возможных результатов работы алгоритма при данном множестве входных данных; определение возможных входных данных, приводящих к данному результату.	25.03.21	12.04.21	18.03.21	05.04.21
27.	4.5	Программирование разветвляющихся алгоритмов. Условный оператор.	Условный оператор: полная и неполная формы. Пример задачи обработки данных по нахождению минимального и максимального числа из двух, трех, четырех данных чисел. Знакомство с алгоритмами решением этой задачи. Реализации этих алгоритмов в выбранной среде программирования. Запись алгоритмических конструкций в выбранном языке программирования.	4 четверть 15.04.21	26.04	4 четверть 08.04.21	19.04
28.	4.6	Составной оператор. Многообразие способов записи ветвлений.	Пример задачи обработки данных по нахождению всех корней заданного квадратного уравнения. Знакомство с алгоритмами решением этой задачи. Реализации этих алгоритмов в выбранной среде программирования. <i>Примеры записи команд ветвления в различных алгоритмических языках.</i>				

29.	4.7	Программирование циклов с заданным числом повторений.	Программирование циклов с заданным числом повторений. Нахождение суммы элементов данной конечной числовой последовательности. Знакомство с алгоритмами решением этой задачи. Реализации этих алгоритмов в выбранной среде программирования.	29.04	17.05	22.04	03.05
30.	4.8	Программирование циклов с заданным условием продолжения работы.	Программирование циклов с заданным условием продолжения работы. Нахождение суммы элементов данной конечной числовой последовательности.	29.04	17.05	22.04	03.05
31.	4.9	Различные варианты программирования циклического алгоритма.	Запись алгоритмических конструкций в выбранном языке программирования. <i>Примеры записи команд повторения и других конструкций в различных алгоритмических языках. Проверка условия выполнения цикла до начала выполнения тела цикла и после выполнения тела цикла: постусловие и предусловие цикла. Инвариант цикла.</i>	13.05	29.05	06.05	17.05
32.	4.10	Проверочная работа по теме «Начала программирования»		13.05	29.05	06.05	17.05
33.	4.11	Анализ контрольной работы	Анализ контрольной работы. Работа над ошибками. Повторение пройденного материала.	27.05		20.05	29.05
34.	4.12	Повторение		27.05		20.05	29.05

В качестве промежуточной аттестации выбран интегрированный зачет, включающий в себя контрольные работы по темам.

