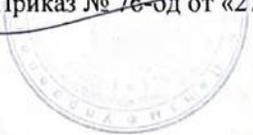


«Утверждаю»  
Директор МБОУ СОШ № 30  
А.А. Долов  
Приказ № 76-од от «27» августа 2020 года



Одобрено педагогическим советом  
МБОУ СОШ № 30 г. Пензы  
Протокол № 1 от «27» августа 2020 года

**Рабочая программа  
по предмету «Компьютерные науки»  
в МБОУ СОШ № 30 г. Пензы  
(11 класс)  
2020-2021 учебный год**

**Учителя: Зайцев В. А., Бычина В. В.**

Рассмотрено на МО  
учителей математики и информатики  
протокол № 1 от «27» августа 2020  
года  
руководитель МО  
  
С.В. Снадина

2020 год

## Пояснительная записка

### Статус документа

Программа первого этапа курса "Компьютерные науки" составлена по рекомендации Министерства образования Пензенской области на основе опыта, накопленного в результате преподавания аналогичного предмета в старших классах системы школьного образования Израиля на протяжении последних 35 лет. Учтены также требования современной сферы рынка высоких технологий (хай-тека) к знаниям выпускников школ, как к потенциальным активным участникам деятельности этой сферы в рамках национальной экономики и международных.

### Структура документа

Рабочая программа включает четыре раздела: пояснительную записку, раскрывающую характеристику и место учебного предмета в учебном плане школы, целей его изучения, основные содержательные линии; основное содержание обучения с распределением учебных часов по разделам и последовательностью изучения тематических блоков в течение года изучения.

### Общая характеристика учебного предмета

Компьютеризация и информатизация стали сегодня естественными составляющими функционирования всех элементов государства и мирового сообщества. Постоянно растущий уровень конкурентности, стремительно увеличивающиеся потребности, практически неограниченное расширение сфер применения компьютерных продуктов и информационных технологий – предъявляют все время растущие требования к выпускнику средней школы. С целью повысить готовность выпускника в минимальные сроки (и даже желательно – на заключительном этапе обучения в школе) стать участником и инноватором государственной и мировой промышленности и экономики – создан курс "Компьютерные науки".

Для достижения учащимися массовой школы образовательных результатов на уровне требований современного мира социальных отношений, государственной жизни, деловой инициативы) необходимы:

- высокий уровень освоения учебных дисциплин, связанных с современными наукоемкими (высокими) технологиями;
- формирование учебных навыков владения методами и инструментами современных технологий, в первую очередь в компьютерной и информационной сфере – не только на уровне пользователя, но и на уровне креативного инноватора;
- проведение учебно-исследовательской и творческо-созидательной работы в технологических средах, отражающей современные реалии экономики и рынка;
- акцент на новых базовых компонентах содержания;
- применение методов работы и методов оценки работы, адекватных новым задачам и перспективам, а также ожидаемым результатам обучения.

Национальная экономика современной России, с постоянно увеличивающейся ролью и долей инновационных технологий, международный динамично развивающийся рынок информационных технологий – демонстрируют непрерывно растущую количественно и все более жесткую качественно потребность в IT-специалистах. При подготовке кадров для данной сферы необходима преэминентность обучения на всех ступенях образования. Стремительно изменяющаяся информационная среда современного общества, научно-технический прогресс подстегивают сферу образования к столь же быстрой модернизации. Выпускник российской школы, решивший стать участником этого процесса, должен сегодня

иметь знания сверх тех, дополнительные к тем, которые он получает в рамках базового курса информатики. Выпускник должен владеть практическими навыками работы с компьютерными и информационными технологиями, быть способным создавать программные и информационные продукты начального (но уже достаточно профессионального) уровня. Эффективное обучение в ВУЗе, успешное трудоустройство и карьерный рост напрямую зависят от степени профессиональных компетенций в сфере компьютерных и информационных технологий.

В 2007 году Ассоциация предприятий компьютерных и информационных технологий (АП КИТ) опубликовала документ «Профессиональные стандарты в области информационных технологий». В документе представлены должностные обязанности, профессиональные компетенции, требования к уровню образования и стажу для девяти наиболее массовых и востребованных профессий в IT-сфере:

1. программист;
2. системный архитектор;
3. специалист по информационным системам;
4. системный аналитик;
5. специалист по системному администрированию;
6. менеджер информационных технологий;
7. менеджер по продажам решений и сложных технических систем;
8. специалист по информационным ресурсам.
9. администратор баз данных.

Анализ этого документа однозначно свидетельствует о том, что сегодня национальная экономика России и международный рынок четко и совершенно однозначно определяют приоритет не только общих фундаментальных знаний выпускников школ и ВУЗов в области алгоритмизации и программировании, знании современных объектно-ориентированных языков и прикладного направления в подготовке будущих специалистов – но и требуют реальной подготовки выпускников в овладении принципиальными (базовыми) и, одновременно, практическими знаниями и навыками.

В существующих программах и УМК по информатике и ИКТ приоритетное внимание отдан освоению офисного пакета программ. Изучаются языки программирования, не предоставляющие прикладного выхода знаний учащихся. Не уделяется внимание самостоятельной проектной работе учащихся.

Данная учебная программа ориентирована на получение одновременно и фундаментальных знаний в области информатики и ИКТ, и на формирование практического навыка применения этих знаний в процессе выполнения учебных проектов в рамках реально используемых в экономике, промышленности и науке современных компьютерных и информационных технологий и подходов.

## **Цели изучения предмета**

### **Основные цели и задачи курса:**

1. формирование у школьников основ научного мировоззрения;
2. обеспечение преемственности между общим и профессиональным образованием;
3. создание условий для самореализации и самовоспитания личности;
4. формирование у школьников базового системного представления о теоретической базе современных компьютерно-информационных технологий;
5. формирование умения креативно и на практике находить эффективные решения исследовательских и практических задач;
6. формирование представления о взаимосвязи и взаимовлиянии современных компьютерных и информационных сфер с фундаментальными и прикладными науками.

## **Место предмета в учебном плане**

В учебном плане МБОУ СОШ № 30 г. Пензы на изучение учебного предмета «Компьютерные науки» в 11 классе отводится 33 часа (1 час в неделю).

## **Общеучебные умения, навыки и способы деятельности**

Образовательная программа предусматривает формирование у учащихся общеучебных умений и навыков, универсальных способов деятельности и ключевых компетенции. В этом направлении приоритетами для учебного предмета «Компьютерные науки» на этапе основного общего образования являются: определение адекватных способов решения учебной задачи на основе заданных алгоритмов; комбинирование известных алгоритмов деятельности в ситуациях, не предполагающих стандартное применение одного из них; использование для решения познавательных и коммуникативных задач различных источников информации, включая энциклопедии, словари, Интернет-ресурсы и базы данных; владение умениями совместной деятельности (согласование и координация деятельности с другими ее участниками; объективное оценивание своего вклада в решение общих задач коллектива; учет особенностей различного ролевого поведения).

## **Результаты обучения**

Обязательные результаты изучения курса «Компьютерные науки» приведены в разделе «Требования к уровню подготовки выпускников», который полностью соответствует стандарту. Требования направлены на реализацию деятельностного и личностно-ориентированного подходов; освоение учащимися интеллектуальной и практической деятельности; овладение знаниями и умениями, необходимыми в повседневной жизни.

Рубрика «Знать/понимать» включает требования к учебному материалу, который усваивается и воспроизводится учащимися. Выпускники должны понимать смысл изучаемых понятий, принципов и закономерностей.

Рубрика «Уметь» включает требования, основанных на более сложных видах деятельности, в том числе творческой: создавать информационные объекты, оперировать ими, оценивать числовые параметры информационных объектов и процессов, приводить примеры практического использования полученных знаний, осуществлять самостоятельный поиск учебной информации. Применять средства информационных технологий для решения задач.

В рубрике «Использовать приобретенные знания и умения в практической деятельности и повседневной жизни» представлены требования, выходящие за рамки конкретного учебного предмета и нацеленные на решение разнообразных жизненных задач.

Основным результатом обучения является достижение базовой информационно-коммуникационной компетентности учащегося.

## **Основное содержание (33 часа)**

### **1. Основные моменты истории программирования (1 час)**

История языков программирования. Классы языков программирования. Машинные языки. Машинные коды. Языки высокого уровня. Ассемблер. Языки Фортран, Паскаль, Бейсик, С.

## **2. Базовая структура программы. Значение комментариев при написании программы. Этапы работы: написание, отладка, прогон (1 час)**

Обязательные элементы программы. Краткие объяснения каждого обязательного элемента. Допустимые изменения. Комментарии и исполняемые части программы. Значение комментариев и правила их написания. Среда программирования. Правила установки и запуска. Основные этапы создания программы. Написание кода и возможные ошибки. Отладка кода и правила исправления ошибок. Сообщения об ошибках. Прогон программы и логические ошибки.

## **3. Переменные и основные типы переменных. Объявление и инициализация переменных. (1 час)**

Смысл понятия "переменная". Цель использования переменных. Переменные и значения (данные). Типы данных и типы переменных. Разнообразие типов данных и переменных. Правило соответствия типа данных и типа переменных – и наоборот. Объявление переменных. Возможные варианты объявления переменных. Инициализация переменных. Возможные варианты инициализации переменных. Возможные ошибки и их исправление.

## **4. Символьные и строковые типы. (1 час)**

Математические и нематематические типы данных и переменных. Особенности нематематических типов. Определение символьного и строкового типов. Объявление и инициализация символьного и строкового типов. Возможное использование этих типов в программах.

## **5. Команды вывода. Команды ввода. (1 час)**

Изменение значения переменных. Понятие ввода данных с клавиатуры. Использование команд ввода для различных типов данных. Команды вывода данных на экран. Два варианта команд вывода данных на экран и особенности их исполнения. Понятие о "дружественном вводе и выводе данных".

## **6. Присвоение данных и инициализация переменных. (2 часа)**

Изменение величины значения и изменение типа значения. Команды изменения типа значений для установления соответствия между типами данных и типами переменных. Операторы выполнения вычислительных (математических) операций и типы данных и переменных. Специальные операторы для "приведения" типов. Возможные ошибки и правила их исправления.

## **7. Операторы «вычисление остатка» и «вычисление частного». (2 часа)**

Операторы «вычисление остатка», «вычисление частного» и типы переменных в вычисляемом выражении. Возможные ошибки – синтаксические и логические. Примеры заданий на использование вычисления результата деления, остатка и частного.

## **8. Оператор "ветвления" (выбора). If с единственным условием, с блоком, со сложным условием (3 часа)**

Последовательный принцип выполнения команд. Принцип выбора в "жизненных" ситуациях и необходимость моделирования выбора в программировании. Понятие ветвления и особенности исполнения команд ветвления. Выбор и условие. Логические выражения и логические значения. Понятие условия. Исполнение выбора в зависимости от образующегося в условии логического значения. Команда выбора в минимальной форме. Условие в команде выбора и правила его записи. Логика исполнения команды выбора в минимальной форме. Использование команд и выражений в операторе выбора. Отработка навыка написания

программ с использованием минимальной и полной форм оператора выбора. Использование в программе нескольких команд выбора. Влияние команд выбора на результаты вывода данных на экран и значения переменных. Возможные синтаксические ошибки и их исправление. Возможные логические ошибки и их устранение.

Понятие блока. Правила написания блока в программе. Использование блока в операторах выбора, правила написания таких блоков. Возможные ошибки и их исправление. Отработка навыков написания классов с использованием команд выбора, содержащих блоки.

Сложное условие и его связь с простым условием. Понятие логических связей между условиями. Логическое "и", логическое "или". Правила написания сложного условия. Приоритеты выполнения логических вычислений. Основные правила логических действий. Возможные синтаксические ошибки при написании сложных условий. Логические ошибки при написании сложных условий.

### **9. Наиболее используемые математические функции. Использование функции random(). Работа со случайными числами. (2 часа)**

Необходимости использования математических и логических действий, которые не всегда можно выразить в виде стандартных выражений. Понятие случайного числа и смысл генерации случайных чисел. Преимущества и недостатки использования генерации случайных чисел как замены вводу данных с клавиатуры. Возможные ошибки (синтаксические и логические) и их исправление.

### **10. Цикл for, использование счетчика, использование сумматора (1 час)**

Необходимость повторных действий в программировании. Ситуации с повторными действиями в повседневной жизни и проблема их моделирования в программировании. Понятие цикла как инструмента повторного исполнения. Возможные варианты повторного исполнения. Понятие о цикле for. Структура цикла. Заголовок цикла и тело цикла. Правила написания заголовка цикла for. Понятие о переменной-счетчике повторов. Варианты написания заголовка цикла for. Особенности написания тела цикла for. Возможные ошибки (синтаксические и логические) при использовании цикла for.

Отработка навыков написания программ с использованием цикла for. Поэтапное усложнения цикла. Особенности написания программ с циклом и командами ввода и вывода. Особенности использования в цикле генерации случайных чисел.

Понятие "сумматора". Примеры использования сумматора в жизни и необходимость моделирования этих ситуаций в программировании. Сходство и отличие между счетчиком случаев и сумматором.

### **11. Цикл for – определение максимума и минимума. (2 часа)**

Понятие максимума и минимума. Примеры использования максимума и минимума в жизни и необходимость моделирования этих ситуаций в программировании. Определение максимума и минимума с помощью оператора выбора. Использование цикла для определения максимума и минимума в серии данных.

### **12. Структура и работа цикла while. (2 часа)**

Понятие цикла с "предусловием" и понятие "гибкого" цикла. Цикл while. Особенности этого цикла. Понятие "условия повтора". Принципиальные различия между циклами for и while. Анализ поставленной задачи с точки зрения выбора подходящего к решению типа цикла. Цикл while и использование логических переменных. Отработка навыков написания программ с использованием цикла while. Возможные синтаксические и логические ошибки и их исправление.

### **13. Вложенные циклы. (2 часа)**

Использование в одной программе нескольких циклов. "Соседние" циклы и "цикл в цикле" (вложенные циклы). Примеры использования вложенных циклов в жизни и необходимость моделирования этих ситуаций в программировании. Правила исполнения вложенных циклов. Возможные синтаксические и логические ошибки и их устранение.

#### **14. Использование пользовательских процедур и функций (2 часа)**

Раздел описания процедур и функций. Различия между понятиями процедуры и функции. Правила описания заголовка процедуры и функции. Локальные переменные и правила их описания и использования. Конфликт между локальными и глобальными переменными с одинаковым именем. Правила описания тела процедуры. Правила описания тела функции и возвращаемый результат. Использование процедур и функций в программе. Возможные синтаксические и логические ошибки и их устранение.

#### **15. Массив – описание, инициализация, свойства. (4 часа)**

Необходимость хранения в программе одновременно большого числа значений. "Стандартная" переменная и переменная с внутренней структурой. Понятие массива. Использование массива для хранения большого числа значений одновременно. Доступ к значениям массива. Описание массива. Инициализация массива. Ввод данных в массив (заполнение массива). Возможные ошибки при использовании массива и их устранение. Понятие длины (размера) массива. Использование массива и циклы.

#### **16. Многомерные массивы. (4 часа)**

Понятие размерности массива. Двумерные массивы и особенности их описания. Доступ к значениям двумерного массива. Матрица. Понятие основной и второстепенной диагонали. Работа с двумерным массивом как с таблицей. Возможные ошибки (синтаксические и логические) и их устранение.

### **УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН**

<b>№ темы</b>	<b>Тема</b>	<b>Количество часов</b>
1.	Основные моменты истории программирования	1
2.	Базовая структура программы. Значение комментариев при написании программы. Этапы работы: написание, отладка, прогон.	1
3.	Переменные и основные типы переменных. Объявление и инициализация переменных.	1
4.	Символьные и строковые типы.	1
5.	Команды вывода. Команды ввода.	1
6.	Присвоение данных и инициализация переменных	2
7.	Операторы «вычисление остатка» и «вычисление частного»	2
8.	Оператор "ветвления" (выбора). If с единственным условием, с блоком, со сложным условием	3
9.	Наиболее используемые математические функции. Использование функции random(). Работа со случайными числами.	2
10.	Цикл for, использование счетчика, использование сумматора	3
11.	Цикл for – определение максимума и минимума.	2
12.	Структура и работа цикла while.	2
13.	Вложенные циклы.	2
14.	Использование пользовательских процедур и функций	2
15.	Массив – описание, инициализация, свойства.	4

16.	Многомерные массивы	4
	Всего:	33

## ТРЕБОВАНИЯ К УРОВНЮ ПОДГОТОВКИ УЧАЩИХСЯ 11 КЛАССА

*В результате изучения Компьютерных наук ученик должен*

Начальный уровень знаний, умений и навыков, необходимых для прохождения данного курса ученик должен **знать/понимать**:

- методы введения, обработки и вывода информации;
- требования к дружелюбности интерфейса современных компьютерно-информационных продуктов;
- основные свойства алгоритма, типы алгоритмических конструкций: следование, ветвление, цикл; понятие вспомогательного алгоритма – и пути их реализации в рамках доступных программно-информационных инструментов и продуктов;
- назначение и функции используемых информационных и коммуникационных технологий;
- уметь:
- выполнять базовые операции над объектами: цепочками символов, числами, списками, деревьями; проверять свойства этих объектов;
- планировать и реализовывать эти операции в виде реального программного продукта;
- планировать, создавать и оперировать информационными объектами, используя графический интерфейс, адаптировать создаваемый продукт к потребностям потенциального клиента, учитывая требования дружелюбности системы "клиент – продукт";

### Результаты изучения учебного курса

**Учащиеся должны знать/понимать/уметь:**

1. Планировать, создавать и отлаживать программы на языке программирования.
2. Создавать программы в соответствии с поставленной общей задачей.
3. Выбирать соответствующие требованиям эффективного решения (создания программы) инструменты языка.
4. Интерпретировать результаты, получаемые в ходе исполнения реальных программ.
5. Оценивать сравнительную эффективность разных вариантов практического решения поставленной задачи (разных программ, достигающих сходного результата).
6. Использовать формальную логику для решения задач практического программирования.
7. Вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний.
8. Структурировать решение, создавая программу на основе методов и подпрограмм.

**Учащиеся должны использовать приобретенные знания и умения в практической деятельности и повседневной жизни:**

1. Пользоваться справочными системами и другими источниками справочной информации. Оценивать достоверность информации.
2. Выбирать адекватные программные и аппаратные средства для решения поставленной задачи.

3. Использовать основные возможности языка программирования и используемого прикладного программного обеспечения – в соответствии с требованиями конкретной поставленной задачи.
4. Диагностировать ошибки программного обеспечения и устранять простейшие неисправности его работы.
5. Использовать системные библиотеки и уже созданные прежде программные компоненты для эффективного конструирования программного продукта для новой задачи.
6. Выполнять требования техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами и инструментами программирования.

### ЛИТЕРАТУРА

1. Бройдо. Вычислительные системы, сети и телекоммуникации – СПб.: Питер, 2002.
2. Блох Д. Эффективное программирование, 2014
3. Мартынов Н.Н. Алгоритмизация и основы объектно-ориентированного программирования на JavaScript. Информатика и ИКТ: профильный уровень. 10-й класс. М.: Бином-Пресс, 2010
4. Федоров А. JavaScript для всех. М.: КомпьютерПресс, 1998.
5. Федоров А. Поискковые системы и каталоги по-русски. КомпьютерПресс, № 2, 1999. С. 36-52.
6. Флэнаган Д. Java. Подробное руководство. СПб: Символ-Плюс, 2008.
7. Хоникатт Д. Использование Интернет. К.: Диалектика, 1997.
8. Хоумер А., Улмен К. Dynamic HTML: справочник. СПб.: Питер, 2000
9. Шилд Г. Java 8. Руководство для начинающих, 2013
10. Янг М. Л. Полное руководство Internet. / Пер. с англ. – Киев: Ирина, ВНУ, 2001

### РЕСУРСЫ ИНТЕРНЕТА

1. CITForum. <http://www.citforum.ru/internet/javascript/>
2. Markup Validation Service <http://validator.w3.org/>
3. Namespaces in XML <http://www.w3.org/TR/REC-xml-names/>
4. Online-справочник <http://javascript.itsoft.ru/>
5. Scriptic <http://www.scriptic.ru/>
6. Specification <http://www.w3.org/TR/REC-html40/> <http://www.w3.org/TR/html401/>
7. SpravkaWeb.Ru <http://www.spravkaweb.ru/main> <http://htmlbook.ru/>
8. Бернс Джо. Самоучитель <http://jsp.newmail.ru/>
9. Библиотека SE@RCHEP <http://www.03www.ru/>
10. Кузина И. А., Храпцов П. Б. <http://webclass.polyn.kiae.su/classes/jsi/first.htm>
11. Основы компьютерных сетей: учебное пособие. М.: Бином, 2006.  
<http://www.microsoft.com/Rus/Msdnaa/Curricula/Default.aspx>
12. Программирование на JavaScript <http://javascript.fud.ru/>
13. Стандарт ECMA-262, 3я редакция.  
<http://www.ecma-international.org/publications/standards/Ecma-262.htm>, 18.06.2009
14. Учебное пособие по HTML  
[http://www.math.spbu.ru/user/mdmitrieva/book\\_html/index.htm](http://www.math.spbu.ru/user/mdmitrieva/book_html/index.htm)
15. Учебное пособие по JavaScript  
[http://www.math.spbu.ru/user/mdmitrieva/book\\_js/index.htm](http://www.math.spbu.ru/user/mdmitrieva/book_js/index.htm)
16. Форум профессиональных web-разработчиков XPoint <http://xpoint.ru/>

**Технические средства обучения и электронные ресурсы,  
используемые на уроках информатики**

ТСО:

1. Компьютеры
2. Проектор
3. Принтер
4. Сканер
5. Web-камеры
6. Наушники
7. Колонки
8. Интерактивная доска Panasonic «Panaboard»
9. Интерактивный проектор Panasonic