

**МБОУ лицей № 73 г.Пензы  
«Лицей информационных систем и технологий»**

Рассмотрена на заседании МО учителей  
математики, физики, информатики  
протокол №1 от 29.08.2016г.  
и рекомендована к рассмотрению на  
педагогическом совете

Председатель МО Пеганова Е.В

Программа рассмотрена и принята на  
педагогическом совете  
протокол №1 от 29.08. 2016г.

**Утверждаю**

приказ № 220-оп от 01.09.2016г.

Директор лицея В.А.Копешкин

**Рабочая программа  
курса по информатике и ИКТ «Компьютерные науки» для учащихся 10-11 классов  
общеобразовательной школы,  
разработанного по заказу Министерства образования Пензенской области  
преподавателями предмета "Компьютерные науки" средней школы "АМИТ - Беэр-Шева"  
г. Беэр-Шева, Израиль Фрайманом В. М и Каннелем Е. Г., учителем информатики МБОУ  
лицей №73 г. Пензы «Лицей информационных систем и технологий»  
Машковой И. Ю., доцентом кафедры информатики и вычислительных систем ПГПУ им.  
В.Г. Белинского Диковым А.В.**

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа первого этапа курса "Компьютерные науки" составлена 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.

### Аннотация

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 преемственность 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 отдано 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.

### **Место и роль курса в обучении**

В настоящее время существуют проблемы, связанные с подготовкой старшеклассников к итоговой государственной аттестации в форме ЕГЭ. Большинство учебных программ по информатике и ИКТ ориентируются на изучение школьниками прикладных программ офисного пакета, на изучение философских и мировоззренческих проблем информационных технологий. Недостаточное внимание уделяется таким важным разделам информатики как теория программирования, основы теории информации, формирование практических навыков создания реальных программных и информационных продуктов, отражающих потребности рынка и науки. Без формирования у выпускника симбиоза глубоких знаний принципов алгоритмизации и программирования и практических креативных навыков создания реальных компьютерно-информационных продуктов – шансы его на успех в качестве эффективного члена и лидера в современном обществе резко снижаются.

Содержание курса дает учащимся и фундаментальное представление о современных подходах к путям и способам практической реализации изучаемой теории, а так же формирует у него навыки практического применения этих представлений в рамках овладения современными компьютерно-информационными технологиями.

Содержание курса удовлетворяет требованиям доступности и понятности материала, его связи с практикой.

### **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.

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";

Курс рассчитан на учащихся 10-11 классов средней школы. На изучение курса отводится 70 часов по 1 учебному часу в неделю. Учебные занятия проводятся в виде лекций, практикумов и контрольных работ. Предусматривается выполнение проектных работ.

Теоретический материал – 50 часов

Практическая работа (в том числе проектная работа) – 20 часов

### **Результаты изучения учебного курса**

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

1. Планировать, создавать и отлаживать программы на языке программирования (Java).
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 (Java) 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.

## УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ темы	Тема	Количество часов	
		теория	практика
1	Основные моменты истории программирования	2	
2	Базовая структура программы на Java Значение комментариев при написании программы Этапы работа: написание, отладка, прогон	2	
3	Переменные и основные типы переменных. Объявление и инициализация переменных	2	
4	Символьные и строковые типы	2	
5	Команды вывода Команды ввода	2	
6	Практикум по темам 2-5		2
7	Присвоение данных между разными типами переменных Инициализация переменных Специальные операторы	2	
8	Операторы «Вычисление остатка» и «вычисление частного»	2	
9	Практикум по теме 8		2
10	Оператор "ветвления" (выбора) – общие соображения	2	
11	Простой if (без else с единственным условием)	2	
12	Простой if (с else с единственным условием)	2	
13	Практикум по темам 11-12		2
14	Простой if с блоком (блоками)	2	
15	Практикум по теме 14		2
16	Оператор if со сложным условием	2	
17	Практикум по теме 16		2
18	Логический тип (булевы значения – «истина» и «ложь»)	2	
19	Наиболее используемые функции библиотеки Math Использование функции random – работа со случайными числами	2	
20	Практикум по теме 19		2
21	Цикл for	2	
22	Цикл for – использование счетчика	2	

23	Цикл for – использование сумматора	2	
24	Цикл for – определение максимума и минимума	2	
25	Практикум по темам 21-24		2
26	Структура и работа цикла while	2	
27	Практикум по теме 26		2
28	Вложенные циклы	2	
29	Использование методов в классах (программах) на языке Java	2	
30	Массив – описание, инициализация, свойства	2	
31	Массивы и методы	2	
32	Массивы счетчиков и массива сумматоров	2	
33	Практикум по темам 29-32		2
34	Итоговая контрольная работа	2	
35	Итоговое повторение		2
Итого		50	20
		70	

## ОСНОВНОЕ СОДЕРЖАНИЕ КУРСА

### Тема №1

#### Основные моменты истории программирования

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

### Тема №2

#### Базовая структура программы на Java

#### Значение комментариев при написании программы

#### Этапы работа: написание, отладка, прогон

Обязательные элементы программы (класса) на языке Java. Краткое объяснения каждого обязательного элемента. Допустимые изменения. Назначения каждой основной части класса. Комментарии и исполняемые части класса. Значение комментариев и правила их написания. Редактор DrJava. Правила установки и запуска. Основные этапы создания класса. Написание кода и возможные ошибки. Отладка кода и правила исправления ошибок. Сообщения об ошибках. Прогон класса и логические ошибки.

### Тема №3

#### Переменные и основные типы переменных.

#### Объявление и инициализация переменных

Смысл понятия "переменная". Цель использования переменных в классе. Переменные и значения (данные). Типы данные и типы переменных. Разнообразие типов данные и переменных. Правило

соответствия типа данных и типа переменных – и наоборот. Объявление переменных. Возможные варианты объявления переменных. Инициализация переменных. Возможные варианты инициализации переменных. Объединение объявления переменных с инициализацией и раздельное выполнение объявления и инициализации. Возможные ошибки и их исправление.

#### **Тема №4**

##### **Символьные и строковые типы**

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 классах (программах).

#### **Тема №5**

##### **Команды вывода**

##### **Команды ввода**

Статический по отношению к исполнению класс. Идея динамичности (интерактивности) связи между переменными и данными. Изменение значения переменных. Понятие ввода данные с клавиатуры. Использование метода Scanner и идеология "переменной-диспетчера". Команды ввода для различных типов данных. Команды вывода данные на экран. Два варианта команд вывода данные на экран и особенности их исполнения. Понятие о "дружественном вводе и выводе данных".

#### **Тема №6.**

##### **Практикум по темам 2-5**

Отработка практических навыков самостоятельного написания, отладки и прогона классов с использованием переменных разных типов, вводом данные разных типов с клавиатуры, вывода данные на экран разными способами. Требование "дружественности" и использование комментариев.

#### **Тема №7**

##### **Присвоение данных между разными типами переменных**

##### **Инициализация переменных**

##### **Специальные операторы**

Возможности изменения значений для занесения их в переменные разных типов. 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.

#### **Тема №8**

##### **Операторы «Вычисление остатка» и «вычисление частного»**

Оператор деления и типы переменных в вычисляемом выражении. Особенности оператора – многозначность и зависимость исполняемой операции от типа переменных. Деление и вычисление остатка – особенности математического смысла и особенности записи оператора в классе. Оператор вычисления частного. Возможные ошибки – синтаксические и логические. Примеры заданий на использование вычисления результата деления, остатка и частного.

#### **Тема №9**

##### **Практикум по теме 8**

Отработка навыков написания классов с использованием деления, вычисления остатка и частного при делении. Задачи из курса математики и их реализация в классах (программах). Задачи общего типа, в которых вычисления результатов деления, значений остатка и частного имеют "общий" смысл.

## **Тема №10**

### **Оператор "ветвления" (выбора) – общие соображения**

Последовательный принцип выполнения команд класса. Принцип выбора в "жизненных" ситуациях и необходимость моделирования выбора в программировании. Понятие ветвления и особенности исполнения команд класса при ветвлении. Выбор и условие. Логические выражения и логические значения. Понятие условия. Исполнение выбора в зависимости от образующегося в условии логического значения.

## **Тема №11**

### **Простой if (без else и с единственным условием)**

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.

## **Тема №12**

### **Простой if (с else и с единственным условием)**

Команда выбора в полной форме. Логика исполнения команды выбора при выполнении условия. Логика выполнения команды выбора при неисполнении условия. Использование команд и выражений в команде выбора в полной форме. Понятие "игнорирования" команд.

## **Тема №13**

### **Практикум по темам 11-12**

Отработка навыка написания классов с использованием минимальной и полной форм оператора выбора. Использование в классе 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.

## **Тема №14**

### **Простой if с блоком (блоками)**

Понятие блока. Правила написания блока в классе (программе). Использование блока в операторах выбора, правила написания таких блоков. Возможные ошибки и их исправление.

## **Тема №15**

### **Практикум по теме 14**

Отработка навыков написания классов с использованием команд выбора, содержащих блоки.

## **Тема №16**

### **Оператор if со сложным условием**

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.

## **Тема №17**

### **Практикум по теме 16**

Отработка навыков написания классов с использованием операторов выбора, содержащих сложные условия. Возможные ошибки и их устранение.

## **Тема №18**

### **Логический тип (булевы значения – «истина» и «ложь»)**

Переменные логического (булевого) типа и значения логического (булевого) типа. Значение "истина" и значение "ложь". Связь между переменными логического типа и математическими выражениями. Использование переменных логического типа. Особенности использования переменных логического типа в командах вывода на экран.



## **Тема №19**

### **Наиболее используемые функции библиотеки Math**

#### **Использование функции `random` – работа со случайными числами**

Необходимость использования математических и логических действий, которые не всегда можно выразить в виде стандартных выражений. Понятие системной библиотеки в языке Java. Системная библиотека Math. Правила использования этой библиотеки. Понятие случайного числа и смысл генерации случайных чисел. Правило генерации случайного числа с использованием системной библиотеки Math. Преимущества и недостатки использования генерации случайных чисел как замены вводу данных с клавиатуры.

## **Тема №20**

### **Практикум по теме 19**

Отработка навыков написания классов с использованием системной библиотеки Math. Отработка навыков написания классов с использованием случайных чисел. Возможные ошибки (синтаксические и логические) и их исправление.

## **Тема №21**

### **Цикл `for`**

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. Особенности описания и инициализации переменной-счетчика повторов. Варианты написания заголовка цикла `for`. Особенности написания тела цикла `for`. Возможные ошибки (синтаксические и логические) при использовании цикла `for`.

## **Тема №22**

### **Цикл `for` – использование счетчика**

Отработка навыков написания классов с использованием цикла `for`. Поэтапное усложнение цикла. Особенности выбора переменной-счетчика повторов. Особенности написания классов с циклом и командами ввода и вывода. Особенности использования в цикле генерации случайных чисел. Понятие "счетчика случаев". Требования к инициализации переменной-счетчику случаев. Использование переменной-счетчика случаев в цикле.

## **Тема №23**

### **Цикл `for` – использование сумматора**

Понятие "сумматора". Примеры использования сумматора в жизни и необходимость моделирования этих ситуаций в программировании. Требования к инициализации переменной-счетчику случаев. Использование переменной-счетчика случаев в цикле. Сходство и отличие между счетчиком случаев и сумматором.

## **Тема №24**

### **Цикл `for` – определение максимума и минимума**

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. Определение максимума и минимума с помощью системной библиотеки Math. Использование цикла для определения максимума и минимума в серии данных.

## **Тема №25**

### **Практикум по темам 21-24**

Отработка навыков написания классов с использованием циклов для выполнения подсчетов (числа случаев и суммы значений) и определения максимума и минимума. Использование в этих классах команд выбора, случайных чисел системной библиотеки Math.

## **Тема №26**

### **Структура и работа цикла while**

Понятие цикла с "предустановленным" числом повторов 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.

## **Тема №27**

### **Практикум по теме 26**

Отработка навыков написания классов с использованием цикла while. Возможные синтаксические и логические ошибки и их исправление.

## **Тема №28**

### **Вложенные циклы**

Использование в одном классе 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.

## **Тема №29**

### **Использование методов в классах (программах) на языке Java**

Главный метод и класс. Команды и главный метод. Дополнительные методы и их связь с главным методом. Понятие метода. Особенности выполнения класса, содержащего методы. Вызов метода (обращение к методу). Передача данных методу. Методы, возвращающие значение и методы, не возвращающие значений. Заголовок метода и тело метода. Правила написания заголовка метода. Команда return в методах, возвращающих значения. Типы методов. Классы с использованием методов и классы без использования методов – сравнение (преимущества и недостатки). Возможные ошибки (синтаксические и логические) и их устранение.

## **Тема №30**

### **Массив – описание, инициализация, свойства**

Необходимость хранения в классе 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 в зависимости от типа данных для хранения. Инициализация массива. Ввод данных в массив (заполнение массива). Возможные ошибки при использовании массива и их устранение. Понятие длины (размера) массива. Использование массива и циклы.

## **Тема №31**

### **Массивы и методы**

Использование методов при работе с массивами. Массив в главном методе и массив в дополнительных методах. Глобальный и локальный массив. Возможные ошибки (синтаксические и логические) и их устранение.

## **Тема №32**

### **Массивы счетчиков и массива сумматоров**

Использование массива для случаев, когда необходимы большое количество счетчиков и сумматоров. Правила использования массива счетчиков и массива сумматоров – в отличие от правил использования массива данных.

## **Тема №33**

### **Практикум по темам 29-32**

Отработка навыков написания классов (программ) с использованием массивов (данных, счетчиков и сумматоров) и всех изученных до этого инструментов (ввода и вывода данных, команд выбора, системной библиотеки Math, случайных чисел, циклов и так далее).