

## АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ «Программирование»

по основной профессиональной образовательной программе по направлению подготовки  
09.03.04 «Программная инженерия» (уровень бакалавриата)

**Направленность (профиль):** Разработка программно-информационных систем

**Трудоемкость дисциплины** – 18 з.е. (648 часов)

**Форма промежуточной аттестации** – Экзамен.

**В результате освоения дисциплины обучающийся должен обладать следующими компетенциями:**

- ОПК-2: Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;
- ОПК-4: Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью;
- ОПК-5: Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;
- ОПК-6: Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов.

**Содержание дисциплины:**

Дисциплина «Программирование» включает в себя следующие разделы:

**Форма обучения очная. Семестр 1.**

**1. Основы программирования на языке СИ.** Современные информационные технологии и программные средства. Основные этапы решения задач на ЭВМ. Нисходящее и восходящее проектирование. Алгоритмы. Интегрированные среды разработки программ. Тестирование и отладка программ. Интерфейс программ. Документирование. Основные элементы Си-программы.

**2. Ввод-вывод в СИ. Операции в СИ.** Основные типы данных в Си. Ввод-вывод в Си. Операции в Си. Приведение типов.

**3. Операторы разветвления.** Условный оператор if. Оператор переключения switch.

**4. Циклические операторы.** Оператор цикла с предусловием while. Оператор цикла с постусловием do...while. Оператор цикла for. Операторы break, continue, goto. Вложенные циклы.

**5. Одномерные массивы.** Основные понятия. Объявление массивов в Си. Типовые задачи по обработке одномерных массивов.

**6. Многомерные массивы.** Двумерные массивы. Типовые задачи по обработке двумерных массивов.

**7. Указатели.** Понятие указателя. Адресные операции. Динамическое распределение памяти. Динамические массивы.

**8. Стандартизация и документирование.** Стандартизация программного обеспечения. Документирование программного обеспечения. Вопросы качества программного обеспечения.

**Форма обучения очная. Семестр 2.**

**1. Строки в СИ.** Понятие строки. Связь массивов, указателей и строк. Массивы строк. Многомерные массивы и указатели.

**2. Стандартные функции для работы со строками.** Стандартные функции. Типовые задачи по обработке строк.

**3. Производные типы (структуры, объединения).** Структуры (записи). Объединения. Переменные структуры. Массивы структур.

**4. Подпрограммы в СИ.** Методология структурного программирования: нисходящее проектирование, модульное программирование, структурное кодирование. Определение и описание функций в Си. Вызов функций. Оператор return. Передача параметров. Передача массивов в качестве параметров. Локальные и глобальные данные.

**5. Функции. Дополнительные возможности.** Рекурсивные функции. Функции с переменным

числом параметров. Функция `main()` – передача параметров и возвращение результата. Передача имени функции в качестве параметра. Указатель на функцию. Классы памяти. Управление видимостью функций.

**6. Файлы.** Ввод-вывод в Си. Файлы (`file`) и потоки (`stream`). Определение потока. Стандартные потоки. Нестандартные потоки. Основные этапы работы с потоком.

**7. Язык СИ++.** Обзор языка Си++. Введение в объектно-ориентированное программирование. Классы и объекты. Инкапсуляция. Наследование. Полиморфизм. Стандартная библиотека С++. Контейнеры и алгоритмы.

**8. Приложения с визуальным графическим интерфейсом.** Форма. Палитра компонентов. Свойства и события. Программы-обработчики событий.

**Форма обучения очная. Семестр 3.**

**1. Введение. Инкапсуляция. Понятие классов и объектов в С++.** Обзор технологий разработки программного обеспечения, отличие процедурного и объектно-ориентированного подходов.

Запись структур в стиле С++. Замена структуры `struct` на класс `class`. Добавление методов в классы. Ограничение доступа (`public:` и `private:`). Общепринятый способ размещения классов по файлам. Динамические объекты на С++.

**2. Язык Java. Инструменты разработки программного обеспечения на языке Java, основные сведения о языке Java.** Инструментальные средства. Язык программирования Java. Консольное приложение на языке Java. Запись классов и объектов на Java. Eclipse и netbeans.

**3. Язык С#. Инструменты разработки программного обеспечения на языке С#, основные сведения о языке С#. Классы и объекты на С#.** Инструментальные средства. Основные сведения о языке С# и NET Framework. Консольное приложение С#. Классы на С#. Структуры на С#.

**4. Объекты в языках С++, Java, С# (Часть1).** Объектно-ориентированная технология разработки программного обеспечения. Вызов метода у объекта класса и внутри класса. Дополнительные сведения о Java и С# (Константы в С++, Java, С#; Пакеты на Java; Массивы объектов на Java и С#; Передача параметров на С++, С# и Java). Перегрузка методов. Указатель `this`. Дружественные функции на С++. Перегрузка операторов на С++ и С#. Работа со строками на С++, Java и С#.

**5. Объекты в языках С++, Java, С# (Часть2).** Объектно-ориентированная технология разработки программного обеспечения. Статические поля и методы на С++, Java и С#. Конструкторы инициализации и деструктор на С++, Java и С#. Конструктор копирования на С++. "Мелкое"(`shallow`) и "глубокое"(`deep`) копирование на С++. Перегрузка оператора присваивания на С++. Исключения на С++, Java и С#.

**6. Наследование. Производные классы в С++, Java, С#.** Объектно-ориентированная технология разработки программного обеспечения. Производный класс, наследование полей и методов и добавление новых полей и методов на С++. Модификатор доступа `protected`. Модификатор в заголовке наследуемого класса. Множественное наследование. Наследование на языке Java. Наследование в языке С#. Перегрузка методов и явное указание класса вызываемого метода. Конструкторы в производных классах. Перегрузка оператора присваивания для производного класса на С++. Библиотека ввода-вывода на С++.

**7. Полиморфизм. Абстрактные классы. Интерфейсы.** Объектно-ориентированная технология разработки программного обеспечения. Виртуальные методы на С++, Java, С#. Полиморфизм. Вызов виртуальной функции на С++ и С# после присваивания указателей. Таблицы виртуальных функций для полиморфных классов. Принципы вызова виртуальных функций. Абстрактные классы на С++, Java, С#. Чисто виртуальные функции С++, абстрактные функции на Java, С#. Интерфейсы на Java и С#. Клонирование объектов на Java и С#. Встроенный интерфейс клонирования. Мелкое и глубокое копирование на Java и С#.

**8. Шаблоны (templates). Обобщенное программирование (Generics).** Объектно-ориентированная технология разработки программного обеспечения. Шаблоны функций (`template`) на С++. Шаблоны классов на С++. Обобщенные методы и классы (`generics`) на Java. Обобщенные методы и классы (`generics`) на С#. Делегаты на С#.

**9. Контейнеры STL библиотеки. Коллекции на Java и С#.** Инструменты разработки ПО на С++, в частности, библиотека STL. Краткие сведения о контейнере `list`, `adapter`, `functor`. Алгоритмы. Контейнер `vector` библиотеки STL с объектами базового и производного классов. Контейнер `map`

библиотеки STL. Инструменты разработки ПО на Java и C# - Коллекции, Обобщенные коллекции.

**10. Примеры ООП. Объектно-ориентированная технология разработки простейших GUI приложений.** Простое GUI приложение на Qt. Простое приложение Windows Forms на C++ и C#. Простое GUI приложение на Java. Простое приложение для Android.

**11. Примеры ООП. Объектно-ориентированная технология разработки GUI приложений..** Все примеры рассматриваются на Windows Forms (C++ и C#), Qt (C++), Java (swing), Android (java). Несколько форм. Меню. Окна диалогов. Чтение и запись файлов. Сериализация. Списки и комбинированные списки, таблицы и пр.

**12. Примеры ООП. Объектно-ориентированная технология разработки GUI приложений (Растровая графика).** Все примеры рассматриваются на Windows Forms (C++ и C#), Qt (C++), Java (swing), Android (java). Классы для работы с графикой. Загрузка изображений. Обработка событий мыши / событий касания экрана. Преобразование фрагмента изображения по отдельным точкам. Алгоритмы масштабирования изображений.

**13. Примеры ООП. Объектно-ориентированная технология разработки GUI приложений (Векторная графика).** Все примеры рассматриваются на Windows Forms (C++ и C#), Qt (C++), Java (swing), Android (java). Рисование геометрических фигур, закрашивание. Использование таймера для анимации изображений.

**14. Примеры ООП. Объектно-ориентированная технология разработки GUI приложений (Обработка текста).** Все примеры рассматриваются на Windows Forms (C++ и C#), Qt (C++), Java (swing), Android (java). Классы для обработки текста. Выбор и установка шрифта. Операции над текстом. Добавка, удаление фрагмента текста. Установка другого шрифта на фрагменте текста. Выравнивание строки. Загрузка текста из файла и сохранение текста. Обработка события нажатия клавиш.

Разработал:  
доцент  
кафедры ПМ  
Проверил:  
Декан ФИТ



Е.В. Егорова

А.С. Авдеев