

## Аннотация рабочей программы дисциплины

**Направление подготовки:** 02.03.02 «Фундаментальная информатика и информационные технологии»

**Тип образовательной программы:** академический бакалавриат

**Направленность (профиль):** Информационная сфера

**Квалификация (степень) выпускника:** бакалавр

**Форма обучения:** очная

### 1. Наименование дисциплины

Б1.Б.17 Информатика и программирование

### 2. Цели и задачи дисциплины (модуля):

Целью изучения дисциплины «Информатика и программирование» является формирование у студентов базовых знаний в области информатики и программирования. Задачами дисциплины «Информатика и программирование» являются формирование знаний, умений и навыков студента по разделам «Введение в программирование», «Представление и обработка чисел в компьютере», «Массивы и строки», «Объектно-ориентированное программирование», «Дополнительные инструменты Java» и «Абстрактные типы данных», формирование практических умений и навыков программирования на языке Java, формирование ключевых профильных компетенций.

### 3. Требования к результатам освоения дисциплины (модуля):

Процесс изучения дисциплины направлен на формирование следующих компетенций:

ОПК-2 – способностью применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий;

ОПК-3 – способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям;

ОПК-4 – способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

В результате изучения дисциплины студент должен:

Знать: основные способы и принципы представления структур данных и приемы алгоритмизации, принципы структурного и модульного программирования, основные приемы алгоритмизации и программирования на языке высокого уровня, принципы разработки программ, принципы автономной отладки и тестирования простых программ, базовые конструкции языка программирования Java.

Уметь: формализовать поставленную задачу, составлять и оформлять программы на языке программирования Java, тестировать и отлаживать программы в современных интегрированных средах разработки, применять полученные знания к различным предметным областям, разрабатывать и описывать классы объектов разного целевого характера, создавать иерархии классов объектов.

Владеть: навыками разработки, отладки и тестирования программ в интегрированной среде разработки, приемами работы с учебной, научной, справочной литературой, навыком распознавания обобщенных приемов и методов решения типовых классов задач.

### 4. Объем дисциплины (модуля) и виды учебной работы

Вид учебной работы	Всего	Семестры
--------------------	-------	----------

	часов / зачетных единиц	1	2
<b>Аудиторные занятия (всего)</b>	154	66	88
В том числе:	–	–	–
Лекции	50	30	20
Практические занятия (ПЗ)	20		20
Лабораторные работы (ЛР)	70	30	40
Контроль самостоятельной работы (КСР)	14	6	8
<b>Самостоятельная работа (всего)</b>	71	33	38
В том числе:	–	–	–
Домашние задания	71	33	54
Вид промежуточной аттестации (зачет, экзамен)		экзамен	экзамен
Общая трудоемкость часы	324	144	180
зачетные единицы	9	4	5

## 5. Краткая характеристика содержания учебной дисциплины

### РАЗДЕЛ 1. ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ

**ТЕМА 1.1.** Языки программирования. Классификация языков программирования  
Язык программирования, системы программирования, транслятор, интерпретатор, компилятор, компоновщик, отладчик, функции и задачи языка программирования, классификация языков программирования, императивные, декларативные, функциональные, логические, объектно-ориентированные языки программирования, языки сценариев, языки параллельных вычислений.

**ТЕМА 1.2.** Алгоритмы и программы

Алгоритм, свойства алгоритма, программа, команда, способы записи алгоритма, основные качества программ, правила написания программного кода, этапы разработки программы.

**ТЕМА 1.3.** Знакомство с языком программирования Java. Переменные

Язык Java, байт-код, виртуальная машина Java, простейшая программа на Java, компиляция и выполнение программы, оператор вывод, переменная, типы переменных, имена переменных, ключевые и зарезервированные слова Java, объявление переменной, оператор присваивания, сокращенная запись арифметических операций, ручная прокрутка программы, порядок выполнения операций, ввод данных с клавиатуры.

**ТЕМА 1.4.** Среда разработки NetBeans IDE. Отладка программы

JDK, IDE – интегрированная среда разработки, NetBeans IDE, установка, создание приложения, создание класса, компиляция и запуск, отладка программы, методы отладки.

**ТЕМА 1.5.** Целые типы данных

Целые типы данных, объявление переменных целого типа, операции над величинами целого типа, особенности деления в Java, определение цифр числа, целочисленные константы.

**ТЕМА 1.6.** Вещественные типы данных. Методы класса Math

Типы с плавающей точкой, константы с плавающей точкой, класс Math, константы класса Math, прямые трансцендентные функции, обратные трансцендентные функции, гиперболические функции, экспоненциальные функции, функции округления, прочие функции, псевдослучайные числа,

**ТЕМА 1.7.** Форматный вывод

Форматный вывод, форматная строка, спецификаторы формата, форматирование целых чисел, форматирование вещественных чисел, спецификатор минимальной ширины поля,

указание точности, флаги формата, выравнивание вывода, использование индекса аргументов.

**ТЕМА 1.8.** Логический тип данных. Условный оператор. Оператор выбора  
Разветвляющиеся алгоритмы, условный оператор, сложные условия, логические операции, порядок выполнения, оператор выбора, выполнение оператора выбора, вложенные операторы выбора.

**ТЕМА 1.9.** Циклические конструкции. Операторы перехода  
Цикл, цикл с известным количеством шагов, цикл с уменьшением циклической переменной, изменение шага циклической переменной, цикл с условием, цикл с постусловием, операторы break, continue, return.

**ТЕМА 1.10.** Статические методы. Рекурсия  
Статические методы, возвращающие и не возвращающие значение, сигнатура метода, описание метода, глобальные и локальные переменные, формальные и фактические параметры, особенности методов, возвращающих значение, логические методы, рекурсия, рекурсивные объекты, рекурсия и математическая индукция, основное правило рекурсии, рекурсия и итерация, вычисление факториала числа, вычисление чисел Фибоначчи, динамическое программирование.

## **РАЗДЕЛ 2. ПРЕДСТАВЛЕНИЕ И ОБРАБОТКА ЧИСЕЛ В КОМПЬЮТЕРЕ**

### **ТЕМА 2.1.** Системы счисления

Системы счисления, перевод чисел из одной системы счисления в другую, перевод дробных чисел из одной системы счисления в другую, понятие экономичности системы счисления, перевод чисел между системами счисления 2-8-16, преобразование нормализованных чисел.

### **ТЕМА 2.2.** Кодирование чисел в компьютере и действия над ними

Кодирование и обработка в компьютере целых чисел без знака, кодирование и обработка в компьютере целых чисел со знаком, кодирование и обработка в компьютере вещественных чисел.

## **РАЗДЕЛ 3. МАССИВЫ И СТРОКИ**

### **ТЕМА 3.1.** Массивы

Одномерный массив, объявление массива, ввод с клавиатуры, поэлементные операции, максимальный элемент массива, заполнение массива случайными числами, реверс массива, циклические сдвиг, сортировка элементов массива, алгоритмы сортировки, метод пузырька, метод пузырька с флажком, метод выбора, формирование массива по условию, поиск в массиве, линейный поиск, двоичный поиск, массив как параметр метода, многомерные массивы (матрицы), объявление многомерного массива, ввод и вывод, обработка всех элементов массива, операции с матрицами, неоднородные (нерегулярные) массивы, инициализация многомерных массивов.

### **ТЕМА 3.2.** Символьный и строковый тип данных

Кодирование символов, кодировка 1 байт на символ, кодировка Unicode, символьный тип данных, массив символов, символьные строки, ввод с клавиатуры, указание кодировки, длина строки, выделение подстроки в строке.

## **РАЗДЕЛ 4. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**

### **ТЕМА 4.1.** Основы объектно-ориентированного программирования

Объектно-ориентированное программирование (ООП), основные принципы ООП, объекты, состояние и поведение объекта, класс, общая форма класса, члены класса, аргументы экземпляра, методы, пример класса Vox, объявление объектов, присваивание переменным объектных ссылок, общая форма объявления метода, вызов метода объекта,

метод с параметрами, конструктор класса, особенности конструкторов, конструктор с параметрами, ключевое слово `this`, сокрытие переменной экземпляра.

#### **ТЕМА 4.2. Методы. Аргументы переменной длины**

Перегрузка методов, перегрузка конструкторов, объекты в качестве параметров, объекты-параметры в конструкторах, способы передачи аргументов, вызов по значению, вызов по ссылке, возврат объектов, введение в управление доступов, спецификаторы доступа, ключевое слово `static`, ограничения на методы `static`, ключевое слово `final`, `varargs` – аргументы переменной длины, задание аргумента переменной длины, особенности методов `varargs`, перегрузка методов `varargs`, `varargs` и неопределенность.

#### **ТЕМА 4.3. Наследование**

Наследование, суперкласс, подкласс, общая форма наследование, доступ к членам класса и наследование, ссылочная переменная суперкласса, ключевое слово `super`, вызов конструктора суперкласса, обращение к члену суперкласса, класс `Object`, методы класса `Object`, порядок вызова конструкторов, конструктор по умолчанию, затенение полей родительского класса, переопределение методов, динамическая диспетчеризация методов, абстрактные методы, абстрактные классы, особенности абстрактных классов, ключевое слово `final` и наследование, предотвращение перегрузки метода, предотвращение наследования класса, пример наследования: абстрактный класс `Triad`, подклассы `Time`, `Date`, класс `Memory`.

#### **ТЕМА 4.4. Принципы объектно-ориентированного программирования**

Ограничение доступа, публичные, закрытые, защищенные члены класса, доступность членов класса, наследование, абстракция данных, инкапсуляция, полиморфизм, достоинства ООП.

#### **ТЕМА 4.5. Интерфейсы**

Интерфейс, определение интерфейса, особенности интерфейсов, реализация интерфейсов, доступ к реализациям через ссылки, частичные реализации, отличия интерфейса от абстрактного класса, пример применения интерфейсов, переменные в интерфейсах, расширение интерфейсов.

#### **ТЕМА 4.6. Пакеты**

Пространство имен, пакеты, простые и составные имена, элементы и имена пакетов, имена классов и интерфейсов, определение пакета, поиск пакета и переменная среда `CLASSPATH`, защита доступа, категории видимости членов класса, доступ к членам класса, импорт пакетов, пакет `java.lang.*`, конфликты имен.

## **РАЗДЕЛ 5. ДОПОЛНИТЕЛЬНЫЕ ИНСТРУМЕНТЫ JAVA**

#### **ТЕМА 5.1. Обработка исключений**

Исключение, исключение `Java`, общая форма обработки исключений, самостоятельная обработка исключений, множественный оператор `catch`, встроенные исключения `Java`.

#### **ТЕМА 5.2. Работа с файлами**

Класс `Scanner`, основы сканирования, примеры сканирования, установка разделителей, класс `FileWriter`.

#### **ТЕМА 5.3. Преобразование типов**

Строго типизированный язык, преобразование типов, неявное преобразование типов, явное преобразование типов, виды приведений, тождественное преобразование, преобразование примитивных типов, расширение простого типа, искажения при расширениях, сужение простого типа, сужение дробного типа до целочисленного, преобразование объектных типов, расширение объектных типов, сужение объектных типов, оператор `instanceof`, преобразование к строке, запрещенные преобразования, применение приведений, автоматическое повышение типов.

#### **ТЕМА 5.4. Оболочки примитивных типов**

Оболочки примитивных типов, класс `Number`, иерархия классов, классы `Double` и `Float`, константы и методы классов `Double` и `Float`, классы `Byte`, `Short`, `Integer` и `Long`, константы

и методы классов Byte, Short, Integer и Long, преобразование строк в числа, преобразование чисел в строки, класс Character, константы и методы класса Character.

#### **ТЕМА 5.5. Битовые операции**

Битовые операции, побитовые операции в Java, побитовые логические операции, инверсия, операция &, маска, обнуление битов, проверка битов, операция |, установка битов в 1, операция ^, инверсия битов, шифровка, битовые сдвиги, логический сдвиг, циклически сдвиг, арифметический сдвиг, автоматическое повышение типов.

### **РАЗДЕЛ 6. АБСТРАКТНЫЕ ТИПЫ ДАННЫХ**

#### **ТЕМА 6.1. Линейные списки, стек, очередь, дек**

Статические и динамические данные, динамические массива, списки, операции со списком, создание узла, добавление узла после заданного, стек, операции со стеком, задача с проверкой расстановки скобок, реализация стека (массив), инфиксная, префиксная, постфиксная форма записи арифметических выражений, вычисление арифметических выражений, очередь, реализация очереди (массив), дек.

#### **ТЕМА 6.2. Классы коллекций**

Коллекция, абстрактный класс AbstractCollection, особенности класса AbstractCollection, методы AbstractCollection, каркас коллекций, ArrayList, LinkedList, Stack, AbstractQueue, ограничение типа сохраняемых объектов.

#### **6. Форма промежуточной аттестации:**

экзамен в 1 и 2 семестрах

#### **7. Разработчик аннотации**

доцент кафедры алгебраических и информационных систем А.С. Зинченко