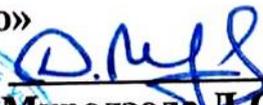


**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ
ТАДЖИКИСТАН
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКО-ТАДЖИКСКИЙ (СЛАВЯНСКИЙ) УНИВЕРСИТЕТ»**

«Утверждаю»
Декан ЕНФ 
Муродзода Д.С.
«31» 08 2024 г.



**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
JAVA – ТЕХНОЛОГИИ**

Направление подготовки: **09.03.03- Прикладная информатика**

Профиль подготовки: **Программная инженерия в экономике**

Форма подготовки: **очная**

Уровень подготовки: **бакалавриат**

Душанбе - 2024

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки РФ № 922 от 19.09.2017 г.

При разработке рабочей программы учитываются

- требования работодателей, профессиональных стандартов по направлению / специальности (при наличии) (для общепрофессиональных и профессиональных дисциплин);
- содержание программ дисциплин/модулей, изучаемых на предыдущих и последующих этапах обучения;
- новейшие достижения в данной предметной области.

Рабочая программа обсуждена на заседании кафедры информатики и ИТ, протокол № 1 от 28 августа 2024 г.

Рабочая программа утверждена УМС естественнонаучного факультета, протокол № 1 от 29 августа 2024 г.

Рабочая программа утверждена Ученым советом естественнонаучного факультета, протокол № 1 от 30 августа 2024 г.

Заведующий кафедрой, к.э.н., доцент



Лешукович А.И.

Зам. председателя УМС факультета
к. ф-м.н., доцент



Халимов И.И.

Разработчик, преподаватель



Каримов М.М.

1. ЦЕЛИ И ЗАДАЧИ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

1.1. Цели изучения дисциплины:

Целями освоения учебной дисциплины «Java технологии» являются:

1. Сформировать системное базовое представление, первичные знания, умения и навыки студентов по основам программирования на объектно-ориентированном языке программирования Java.
2. Подготовить студентов к применению знаний программирования на языке Java в последующих дисциплинах, в обучении в магистратуре, а также после окончания обучения в профессиональной деятельности.

1.2. Задачи изучения дисциплины:

Задачи дисциплины формулируются в соответствии с требованиями ФГОС, предъявляемые к компетенциям обучающегося. В результате освоения дисциплины Java технологии формируются определенный набор компетенции обучающегося:

- 1.3. В результате изучения данной дисциплины у обучающихся формируются следующие общепрофессиональные/ профессиональные / профессионально-специализированные, профессионально-дополнительные компетенции (элементы компетенций)

Таблица 1.

1) Профессиональные компетенции: проектная деятельность:

Код	Формируемая компетенция	Содержание этапа формирования компетенции
ПК-2	Способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение	ПК-2.1 Знать: - основные этапы и принципы создания программного продукта принципы, базовые концепции технологий программирования; характерные особенности и возможности среды разработки приложений Java; основные сведения о процессоре электронных таблиц Excel
		ПК-2.2 Уметь: - составлять алгоритмы решения задач различной структуры и оформлять их в соответствии с синтаксическими правилами языка программирования Java; разрабатывать пользовательский интерфейс приложения, обеспечивающий оптимальное функционирование программы
		ПК-2.3 Владеть: средствами для разработки веб-приложений.

Примечание: Виды оценочных средств необходимо определить в соответствии с Таблицей 5 Раздела 4 рабочей программы дисциплины и Фондом оценочных средств (далее - ФОС) по дисциплине.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Для изучения данной дисциплины студент должен получить необходимые знания, умения и компетенции, которые формируются в результате изучения перечисленных ниже дисциплин: «Основы информатики», «Математическая логика и теория алгоритмов», «Языки и методы программирования», «Операционные системы», «Алгоритмы. Построение и анализ», «Иностранный язык», «Проектирование программного обеспечения», «Объектно-ориентированное программирование».

Курс рассчитан на студентов, имеющих подготовку по предшествующим курсам, касающихся основ программирования с использованием процедурных и объектно-ориентированных методологий (алгоритмический язык Си). В течение преподавания курса предполагается, что студенты знакомы также с теорией создания база данных и структурированным языком запросов SQL, которые преподаются на факультете в рамках курса «Базы данных» перед изучением данной дисциплины.

В свою очередь, данный курс, помимо самостоятельного значения, является предшествующей дисциплиной для курсов: «CASE-средства проектирования программного обеспечения», «Технологии параллельного программирования», «Веб-приложения на Java», «Сетевые технологии и сетевое программирование», «Программная реализация экспертных систем и генетических алгоритмов», «Системное программирование».

Кроме этого знания и умения, полученные в результате освоения данной дисциплины могут быть использованы при прохождении преддипломная практики, подготовке им выпускной дипломной работы, а также в научной и практической деятельности после окончания университета.

Таблица 2.

дисциплины в таблице указаны в качестве примера

№	Название дисциплины	Семестр	Место дисциплины в структуре ООП
1.	<i>Математика</i>	1-2	Б1.О.07
2.	<i>Дискретная математика</i>	1	Б1.О.08
3.	<i>Теория алгоритмов</i>	2	Б1.О.11
4.	<i>Теория вероятности и математическая статистика</i>	2	Б1.О.09
5.	<i>Операционные системы</i>	2	Б1.О.10
6.	<i>Практикум по программированию</i>	2-4	Б1.О.14
7.	<i>Численные методы</i>	3	Б1.В.09
8.	<i>Базы данных</i>	3-4	Б1.О.15
9.	<i>Вычислительные системы сети и телекоммуникации</i>	3-4	Б1.О.16
10.	<i>Разработка программных приложений/ Системное и прикладное программное обеспечение</i>	4	Б1.В.11
11.	<i>Программная инженерия</i>	4-5	Б1.О.17
12.	<i>Информационная безопасность</i>	5	Б1.О.18
13.	<i>Интеллектуальные информационные системы</i>	5	Б1.В.12
14.	<i>Программирование в среде Delphi</i>	7	Б1.В.ДВ.01.01
15.	<i>Программирование в среде Borland C++</i>	7	Б1.В.ДВ.01.20
16.	<i>Интернет-программирование</i>	7	Б1.В.ДВ.02.01
17.	<i>Управление программными проектами</i>	7-8	Б1.В.20
18.	<i>Учебно-вычислительная практика</i>	4	Б2.0.01(У)
19.	<i>Производственная практика</i>	8	Б2.0.02 (П)
20.	<i>Преддипломная практика</i>	8	Б2.В.01 (ПД)

(дисциплины в таблице необходимо указать в соответствии с учебным планом года начала подготовки направления/специальности по которому осуществляется обучение)

3. СТРУКТУРА И СОДЕРЖАНИЕ КУРСА, КРИТЕРИИ НАЧИСЛЕНИЯ БАЛЛОВ

Объем дисциплины (модуля) составляет 6 зачетных единиц, всего 216 часов, из которых: лекции 16 час., практические занятия 36 час., лабораторные работы 36 час., КСР

16 час., всего часов аудиторной нагрузки 104 час., в том числе всего часов в интерактивной форме 26 час., самостоятельная работа 66 час. экзамен 7 семестр.

3.1 Структура и содержание теоретической части курса

Содержание теоретической части курса разбивается на разделы, темы.

Тема 1. Краткий обзор Java. История и развитие языка Java. Идентификаторы. Литералы. Комментарии. Типы данных, переменные и константы.

Тема 2. Операции. Арифметические операции. Поразрядные операции. Операции отношения. Логические операции. Операция присваивания. Предшествование операций. Применение круглых скобок.

Тема 3. Управляющие операторы. Операторы выбора (case). Операторы цикла (for, while, do-while). Операторы перехода (break, continue, return).

Тема 4. Введение в классы. Основы классов. Объявление объектов. Присваивание переменным ссылок на объекты. Введение в методы. Конструкторы. Ключевое слово this. Сборка "мусора". Метод finalize (). Класс Stack.

Тема 5. Подробное рассмотрение классов и методов. Перегрузка методов. Применение объектов в качестве параметров. Подробное рассмотрение особенностей передачи аргументов. Возврат объектов. Рекурсия. Введение в управление доступом. Ключевое слово static. Ключевое слово final. Вложенные и внутренние классы. Применение аргументов командной строки. Аргументы переменной длины.

Тема 6. Массивы и матрицы. Формирование массива и вывод его элементов. Анализ элементов массива. Работа с несколькими массивами. Преобразование массива. Формирование матрицы и вывод ее элементов. Анализ элементов матрицы. Преобразование матрицы. Диагонали квадратной матрицы.

Тема 7. Обработка символьных строк. Конструкторы символьных строк. Длина символьной строки. Специальные строковые операции. Извлечение символов. Сравнение символьных строк. Поиск в символьных строках. Видоизменение символьных строк. Преобразование данных методом valueOf. Изменение регистра символов в строке. Соединение символьных строк. Дополнительные методы из класса String. Класс StringBuffer. Класс StringBuilder.

Тема 8. Пакет java.io для ввода-вывода. Классы и интерфейсы ввода-вывода. Класс File. Интерфейсы AutoCloseable, Closeable и Flushable. Исключения ввода-вывода. Два способа закрытия потока ввода-вывода. Классы потоков ввода-вывода. Потоки ввода-вывода байтов. Потоки ввода-вывода символов. Преимущества потоков ввода-вывода.

Тема 9. Введение в библиотеку Swing (JavaFx). Происхождение библиотеки Swing. Главные особенности библиотеки Swing. Компоненты и контейнеры. Классы JLabel и ImageIcon. Класс JTextField. Кнопки из библиотеки Swing. Класс JList. Класс JComboBox. Класс JTable. Введение в меню Swing. Интерфейс JDBC.

3.2 Структура и содержание практической части курса

Структура и содержание практической части курса включает в себя тематику и содержание практических занятий, семинаров, лабораторных работ.

Практические занятия (36 час.)

1. Операторы.
2. Управляющие конструкции.
3. Массивы.
4. Абстракция.
5. Объекты.
6. Абстрактные классы.
7. Интерфейсы.
8. Управление доступом.
9. Инкапсуляция.

10. Наследование и полиморфизм.
11. Коллекции объектов.
12. Система ввода-вывода Java.
13. Интерфейсы Observable, Iterable, Comparable, Cloneable
14. События и их слушатели (ActionListeners)
15. Библиотека Swing.
16. Model-View-Controller.
17. Диспетчеры компоновки.
18. Библиотека SWT.

Лабораторные работы (36 час.)

1. Ввод и вывод данных, оператор присваивания: группа Begin
2. Целые числа: группа Integer
3. Логические выражения: группа Boolean
4. Условный оператор: группа If
5. Оператор выбора: группа Case
6. Цикл с параметром: группа For
7. Цикл с условием: группа While
8. Последовательности: группа Series
9. Процедуры и функции: группа Proc
10. Минимумы и максимумы: группа Minmax
11. Одномерные массивы: группа Array
12. Двумерные массивы (матрицы): группа Matrix
13. Символы и строки: группа String
14. Двоичные (типизированные) файлы: группа File
15. Символьные и строковые файлы
16. Текстовые файлы: группа Text
17. Составные типы данных в процедурах и функциях: группа Param
18. Рекурсия: группа Recur

3.3 Структура и содержание КСР

1. Массивы.
2. Абстрактные классы.
3. Внутренние и анонимные (безымянные) внутренние классы.
4. Интерфейсы Observable, Iterable, Comparable, Cloneable
5. События и их слушатели (ActionListeners)
6. Библиотека Swing.
7. Model-View-Controller.
8. Диспетчеры компоновки.
9. Библиотека SWT.

Таблица 3.

№ п/п	Раздел дисциплины	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Лит ерат ура	Кол-во баллов в недел ю
		Лек.	Пр.	Лаб.	КСР	СРС		
Семестр 7								
1.	Тема 1. Краткий обзор Java. История и развитие языка Java. Идентификаторы. Литералы. Комментарии. Типы данных, переменные и константы.	2	4	4	2	12	1,3,4 7,8, 21	3

	<p>Лабораторная работа</p> <p>Выполнение и подготовка к защите работы 1</p> <p>Практическое занятие</p>							
2.	<p>Тема 2. Операции. Арифметические операции. Поразрядные операции. Операции отношения. Логические операции. Операция присваивания. Предшествование операций. Применение круглых скобок.</p> <p>Лабораторная работа</p> <p>Практическое занятие Групповые консультации по темам 1-2</p>	2	4	4	2	12	1,3,4 7,8, 21	3
...	<p>Тема 3. Управляющие операторы. Операторы выбора (case). Операторы цикла (for, while, do-while). Операторы перехода (break, continue, return).</p> <p>Лабораторная работа</p> <p>Выполнение и подготовка к защите работы 2</p> <p>Групповые консультации по теме 3</p>	2	4	4	2	12	1,3,4 7,8, 21	3
18.	<p>Тема 4. Введение в классы. Основы классов. Объявление объектов. Присваивание переменным ссылок на объекты. Введение в методы. Конструкторы. Ключевое слово this. Сборка "мусора". Метод finalize (). Класс Stack.</p> <p>Практическое занятие Групповые консультации по теме 4</p>	2	4	4	2	12	1,3,4 7,8, 21	3
	<p>Тема 5. Подробное рассмотрение классов и методов. Перегрузка методов. Применение объектов в качестве параметров. Подробное рассмотрение особенностей передачи аргументов. Возврат объектов. Рекурсия. Введение в управление доступом. Ключевое слово static. Ключевое слово final. Вложенные и внутренние классы. Применение аргументов командной строки. Аргументы переменной длины.</p> <p>Лабораторная работа</p> <p>Выполнение и подготовка к защите работы 3</p> <p>Групповые консультации по темам 5</p>	2	4	4	2	12	1,3,4 7,8, 21	3

	<p>Тема 6. Массивы и матрицы. Формирование массива и вывод его элементов. Анализ элементов массива. Работа с несколькими массивами. Преобразование массива. Формирование матрицы и вывод ее элементов. Анализ элементов матрицы. Преобразование матрицы. Диагонали квадратной матрицы.</p> <p>Лабораторная работа</p> <p>Выполнение и подготовка к защите работы 4</p> <p>Групповые консультации по теме 6</p>	2	4	4	2	12	1,3,4 7,8, 21	3
	<p>Тема 7. Обработка символьных строк. Конструкторы символьных строк. Длина символьной строки. Специальные строковые операции. Извлечение символов. Сравнение символьных строк. Поиск в символьных строках. Видоизменение символьных строк. Преобразование данных методом valueOf. Изменение регистра символов в строке. Соединение символьных строк. Дополнительные методы из класса String. Класс StringBuffer. Класс StringBuilder.</p> <p>Лабораторная работа</p> <p>Выполнение и подготовка к защите работы 5</p> <p>Практическое занятие</p>	2	4	4	2	12	1,3,4 7,8, 21	3
	<p>Тема 8. Пакет java.io для ввода-вывода. Классы и интерфейсы ввода-вывода. Класс File. Интерфейсы AutoCloseable, Closeable и Flushable. Исключения ввода-вывода. Два способа закрытия потока ввода-вывода. Классы потоков ввода-вывода. Потоки ввода-вывода байтов. Потоки ввода-вывода символов. Преимущества потоков ввода-вывода.</p> <p>Лабораторная работа</p> <p>Выполнение и подготовка к защите работы 5</p> <p>Групповые консультации по темам 7-8</p>	2	4	4	2	12	1,3,4 7,8, 21	3
	<p>Тема 9. Введение в библиотеку Swing (JavaFx). Происхождение библиотеки Swing. Главные особенности библиотеки Swing. Компоненты и контейнеры. Классы JLabel и ImageIcon. Класс JTextField. Кнопки из библиотеки Swing. Класс JList. Класс</p>	2	4	4	2	12	1,3,4 7,8, 21	3

JComboBox. Класс JTable. Введение в меню Swing. Интерфейс JDBC. Лабораторная работа Практическое занятие								
ИТОГО: ВСЕГО-216	18	36	36	18	108			

Формы контроля и критерии начисления баллов

Контроль усвоения студентом каждой темы осуществляется в рамках балльно-рейтинговой системы (БРС), включающей текущий, рубежный и итоговый контроль. Студенты **1 курсов**, обучающиеся по кредитно-рейтинговой системе обучения, могут получить максимально возможное количество баллов - 300. Из них на текущий и рубежный контроль выделяется 200 баллов или 49% от общего количества.

На итоговый контроль знаний студентов выделяется 51% или 100 баллов. Из них 16 баллов администрацией могут быть представлены студенту за особые заслуги (призовые места в Олимпиадах, конкурсах, спортивных соревнованиях, выполнение специальных заданий, активное участие в общественной жизни университета).

Порядок выставления баллов: 1-й рейтинг (1-9 неделя по 11,5 баллов = 8 баллов административных, итого 100 баллов), 2-й рейтинг (10-18 неделя по 11,5 баллов = 8 баллов административных, итого 100 баллов), итоговый контроль 100 баллов.

К примеру, за текущий и 1-й рубежный контроль выставляется 100 баллов: лекционные занятия – 20 баллов, за практические занятия (КСР, лабораторные) – 32 балла, за СРС – 20 баллов, требования ВУЗа – 20 баллов, административные баллы – 8 баллов.

В случае пропуска студентом занятий по уважительной причине (при наличии подтверждающего документа) в период академической недели, деканат факультета обращается к проректору по учебной работе с представлением об отработке студентом баллов за пропущенные дни по каждой отдельной дисциплине с последующим внесением их в электронный журнал.

Итоговая форма контроля по дисциплине (зачет, зачет с оценкой, экзамен) проводится как в форме тестирования, так и в традиционной (устной) форме. Тестовая форма итогового контроля по дисциплине предусматривает: для естественнонаучных направлений – 10 тестовых вопросов на одного студента, где правильный ответ оценивается в 10 баллов, для гуманитарных направлений/специальности – 25 тестовых вопросов, где правильный ответ оценивается в 4 балла. Тестирование проводится в электронном виде, устный экзамен на бумажном носителе с выставлением оценки в ведомости по аналогичной системе с тестированием.

Таблица 4.

Неделя	Активное участие на лекционных занятиях, написание конспекта и выполнение других видов работ*	Активное участие на практических (семинарских) занятиях, КСР	СРС Написание реферата, доклада, эссе Выполнение других видов работ	Административный балл за примерное поведение	Балл за рубежный и итоговый контроль	Всего
1	2	3	4	5	6	7
1	-	-	-	-	-	-
2	1	1	1	-	-	3
3	1	1	1	-	-	3

4	1	1	1	-	-	3
5	1	1	1	-	-	3
6	1	1	1	-	-	3
7	1	1	1	-	-	3
8	1	1	1	-	-	3
9 (первый рубежный контроль)					10	10
Первый рейтинг	7	7	7	-	10	31
10	1	1	1	-	-	3
11	1	1	1	-	-	3
12	1	1	1	-	-	3
13	1	1	1	-	-	3
14	1	1	1	-	-	3
15	1	1	1	-	-	3
16	1	1	1	-	-	3
17	1	1	1	-	-	3
18 (второй рубежный контроль)					10	10
Второй рейтинг	8	8	8	5	10	39
ИТОГОВЫЙ КОНТРОЛЬ (зачет, зачет с оценкой, экзамен)					30	30
ИТОГО:	15	15	15	5	20+30	100

***Примечание:** в случае отсутствия лекционных занятий по дисциплине, баллы начисляются за активное участие в практических (семинарских) занятиях, КСР (см. графы 2 и 3 Таблицы с баллами).

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Данный курс предусматривает изучение теоретических вопросов, выполнение лабораторных работ и КСР.

В теоретической части курса изучаются базовые понятия по вопросам объектно-ориентированного программирования, проектирования классов и интерфейсов, конструированию внутренних локальных классов, созданию и управлению объектами, построению консольных и графических приложений и работе с потоками ввода-вывода.

В практической части курса изучается синтаксис языка программирования, правила реализации классов и методов, типы данных, обработка событий и реализация интерфейсов, создание консольных программ, апплетов и фреймов.

По окончании курса студент должен знать синтаксис языка Java, понимать структуру Java-пакетов и кода Java-программы средней сложности; знать основные возможности платформы Java (Standard Edition, Enterprise Edition, Micro Edition), объектно-ориентированную методологию программирования и модель языка Java, знать принципы создания многопоточных приложений и GUI-интерфейсов, технологию JDBC для создания подключения к базе данных.

Для изучения теоретической части курса необходимо изучить вопросы, рассматриваемые в лекциях. При изучении материала необходимо помимо лекционных материалов использовать рекомендуемую литературу для лучшего усвоения материала. Кроме этого, все необходимые сведения, касающиеся конкретных вопросов программирования на Java можно получить из открытых источников, например <http://www.intuit.ru>, <http://www.citforum.ru>, <http://www.opennet.ru>. В конце каждой лекции предлагаются вопросы и задания для самостоятельного изучения с целью углубленного изучения дисциплины.

На лабораторных занятиях с использованием средств вычислительной техники студенты выполняют задания, предусмотренные для приобретения пользовательских навыков, решают задачи вычислительного характера, устанавливают и настраивают программные продукты, разрабатывают алгоритмы и программы для решения прикладных задач, выполняют типовые расчеты. Во время самостоятельной работы студента с преподавателем обсуждаются проблемные лекции, решаются сложные алгоритмы.

1. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

В данном разделе РПД приводится перечень основной литературы (учебники, учебные пособия, монографии) и перечень дополнительной литературы, в который включаются издания, рекомендуемые для углубленного изучения. В перечень основной литературы должны входить учебники, учебные пособия и монографии, изданные в течение последних 5 лет для гуманитарных, социальных и экономических дисциплин и 10 лет для технических, математических и естественнонаучных дисциплин. Не менее трех источников основной литературы, указанных в РПД, должны быть доступны обучающимся в одной или нескольких электронно- библиотечных системах (электронных библиотеках), сформированных на основании прямых договорных отношений с правообладателями. В данном случае необходимо привести полное библиографическое описание источника и рабочую гиперссылку на соответствующий электронный ресурс. В список основной литературы также могут быть включены печатные издания, имеющиеся в фондах РТСУ в количестве, предусмотренном соответствующим ФГОС ВО

5.1 Основная литература

1. *Зыков, С. В.* Программирование. Объектно-ориентированный подход [Текст]: учебник и практикум для академического бакалавриата / С. В. Зыков. — Москва: Издательство Юрайт, 2019. — 155 с.
2. *Тузовский, А. Ф.* Объектно-ориентированное программирование [Текст]: учебное пособие для прикладного бакалавриата / А. Ф. Тузовский. — Москва: Издательство Юрайт, 2019. — 206 с.
3. *Огнева, М. В.* Программирование на языке C++: практический курс [Электронный ресурс]: учебное пособие для среднего профессионального образования / М. В. Огнева, Е. В. Кудрина. — Москва: Издательство Юрайт, 2019. — 335 с. — URL: <https://biblio-online.ru/bcode/439046>.
4. *Черпаков, И. В.* Основы программирования [Электронный ресурс]: учебник и практикум для прикладного бакалавриата / И. В. Черпаков. — Москва: Издательство Юрайт, 2019. — 219 с. — URL: <https://biblio-online.ru/bcode/433423>.
5. *Лаврищева, Е. М.* Программная инженерия. Парадигмы, технологии и CASE-средства [Электронный ресурс]: учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. — Москва: Издательство Юрайт, 2020. — 280 с. — URL: <http://biblio-online.ru/bcode/452156>.

5.2 Дополнительная литература

1. *Подбельский, В. В.* Программирование. Базовый курс C# [Электронный ресурс]: учебник для среднего профессионального образования / В. В. Подбельский. —

Москва: Издательство Юрайт, 2019. — 369 с.— URL: <https://biblio-online.ru/bcode/445334>.

2. *Маркин, А. В.* Программирование на SQL в 2 ч. Часть 1 [Электронный ресурс]: учебник и практикум для бакалавриата и магистратуры / А. В. Маркин. — Москва: Издательство Юрайт, 2019. — 362 с.— URL: <https://biblio-online.ru/bcode/437153>.
3. *Кувшинов, Д. Р.* Основы программирования [Электронный ресурс]: учебное пособие для вузов / Д. Р. Кувшинов. — Москва: Издательство Юрайт, 2019. — 104 с.— URL: <https://biblio-online.ru/bcode/441475>
4. *Трофимов, В. В.* Алгоритмизация и программирование [Электронный ресурс]: учебник для академического бакалавриата / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва: Издательство Юрайт, 2019. — 137 с. — URL: <https://biblio-online.ru/bcode/423824>

5.3 Нормативно-правовые материалы (по мере необходимости)

5.4 Перечень ресурсов информационно-телекоммуникационной сети Интернет

В данном разделе приводится перечень ресурсов информационно телекоммуникационной сети «Интернет», необходимых для освоения дисциплины, в виде названия сайта, интернет - портала и т.п. и рабочей гиперссылки. Не допускается размещение ресурсов, содержащих материалы, несоответствующие этическим нормам, в том числе в формате баннеров и т.п.

1. [Sun Microsystems, Inc. JDK 6 Documentation - Режим доступа: http://java.sun.com/javase/6/docs/www.osborne.com](http://java.sun.com/javase/6/docs/www.osborne.com)
2. <https://habrahabr.ru>
3. <https://www.java.com/ru>
4. www.ibm.com/developerworks/ru
5. <https://info.javarush.ru/>
6. <https://students.uni-vologda.ac.ru>
7. <https://lifehacker.ru>
8. <https://javabegin.ru>
9. <https://biblio-online.ru/>

Перечень информационных технологий и программного обеспечения

Используются лицензионное программное обеспечение ОС Windows -7 и программное обеспечение открытого доступа (Open source) среда NetBeans (Eclipse).

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Данный курс предусматривает изучение теоретических вопросов, выполнение лабораторных работ и КСР.

В теоретической части курса изучаются базовые понятия по вопросам объектно-ориентированного программирования, проектирования классов и интерфейсов, конструированию внутренних локальных классов, созданию и управлению объектами, построению консольных и графических приложений и работе с потоками ввода-вывода.

В практической части курса изучается синтаксис языка программирования, правила реализации классов и методов, типы данных, обработка событий и реализация интерфейсов, создание консольных программ, апплетов и фреймов.

По окончании курса студент должен знать синтаксис языка Java, понимать структуру Java-пакетов и кода Java-программы средней сложности; знать основные возможности платформы Java (Standard Edition, Enterprise Edition, Micro Edition), объектно-ориентированную методологию программирования и модель языка Java, знать принципы создания многопоточных приложений и GUI-интерфейсов, технологию JDBC для создания подключения к базе данных.

Для изучения теоретической части курса необходимо изучить вопросы, рассматриваемые в лекциях. При изучении материала необходимо помимо лекционных материалов использовать рекомендуемую литературу для лучшего усвоения материала. Кроме этого, все необходимые сведения, касающиеся конкретных вопросов программирования на Java можно получить из открытых источников, например

<http://www.intuit.ru>, <http://www.citforum.ru>, <http://www.opennet.ru>. В конце каждой лекции предлагаются вопросы и задания для самостоятельного изучения с целью углубленного изучения дисциплины.

На лабораторных занятиях с использованием средств вычислительной техники студенты выполняют задания, предусмотренные для приобретения пользовательских навыков, решают задачи вычислительного характера, устанавливают и настраивают программные продукты, разрабатывают алгоритмы и программы для решения прикладных задач, выполняют типовые расчеты. Во время самостоятельной работы студента с преподавателем обсуждаются проблемные лекции, решаются сложные алгоритмы.

Все это может дать положительный результат, если студент активно занимается самостоятельной работой в соответствии с планом-графиком п.4.1. Процесс выполнения СРС с описанием этапов решения примерной задачи описан в п.4.2

Вместе с тем основой обучения являются аудиторские занятия - лекции, практические занятия и лабораторные работы по рассмотрению проблем информационной технологии и решению конкретного прикладного задача. Поэтому рассмотрим каждую тему отдельно, чтобы указать на какие моменты обратить внимания, чтобы лучше освоить материал темы.

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации дисциплины при кафедре информатики и ИС РТСУ имеются 5 компьютерных классов, 2 из которых обеспечены электронными досками. В трех компьютерных классах реализованы облачные технологии на базе блейд-серверной системы

8. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Форма итоговой аттестации: экзамен.

Итоговая система оценок по кредитно-рейтинговой системе с использованием буквенных символов

Оценка по буквенной системе	Диапазон соответствующих наборных баллов	Численное выражение оценочного балла	Оценка по традиционной системе
A	10	95-100	Отлично
A-	9	90-94	
B+	8	85-89	Хорошо
B	7	80-84	
B-	6	75-79	
C+	5	70-74	Удовлетворительно
C	4	65-69	
C-	3	60-64	
D+	2	55-59	
D	1	50-54	Неудовлетворительно
Fx	0	45-49	
F	0	0-44	

Содержание текущего контроля, промежуточной аттестации, итогового контроля раскрываются в фонде оценочных средств, предназначенных для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС ВО.

ФОС по дисциплине является логическим продолжением рабочей программы учебной дисциплины. ФОС по дисциплине прилагается.

