

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

УТВЕРЖДАЮ

Проректор по учебной работе
учреждения образования «Белорусский
государственный университет
информатики и радиоэлектроники»

_____ В.А. Рыбак

25.09.2024

Регистрационный № 5-030/уч.

«ПРОГРАММИРОВАНИЕ НА ЯЗЫКАХ ВЫСОКОГО УРОВНЯ»

**Учебная программа учреждения образования по учебной дисциплине
для специальности**

6-05-0611-05 Компьютерная инженерия

2024 г.

Учебная программа учреждения образования составлена на основе примерной учебной программы по учебной дисциплине «Программирования на языках высокого уровня», утвержденной Министерством образования Республики Беларусь 27.06.2024, регистрационный номер № 6-05-06-062/пр. и учебных планов специальности 6-05-0611-05 «Компьютерная инженерия».

СОСТАВИТЕЛИ:

А.М.Ковальчук, старший преподаватель кафедры электронных вычислительных машин учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»;

Д.Н.Марченков, системный архитектор иностранного общества с ограниченной ответственностью «ЭПАМ Системз»;

Д.Ю.Перцев, доцент кафедры электронных вычислительных машин учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент;

В.А.Сицко, старший преподаватель кафедры информационных систем и технологий Института информационных технологий учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»

РАССМОТРЕНА И РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ

Кафедрой электронных вычислительных машин учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 18 от 06.05.2024);

Научно-методическим советом учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 1 от 20.09.2024).

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа рассчитана на 336 учебных часов (9 з.е.)

План учебной дисциплины в дневной форме получения образования:

Код специальности	Название специальности	Курс	Семестр	Аудиторных часов (в соответствии с учебным планом уо)					Академ. часов на курс. работу (проект)	Форма промежуточной аттестации
				Всего	Лекции	Лабораторные занятия	Практические занятия	Семинарские занятия		
6-05-0611-05	Компьютерная инженерия	2	3	84	52	32/24*	-/8*	-	40	зачет
		2	4	66	34	32/24*	-/8*	-		экзамен
	Итого			150	86	64/48*	-/16*	-	40	

Примечание: * - только для профилизации «Компьютерные системы и сети»

План учебной дисциплины в заочной форме получения образования:

Код специальности	Название специальности	Курс	Семестр	Аудиторных часов (в соответствии с учебным планом уо)				Академ. часов на курс. работу (проект)	Контрольные работы	Форма промежуточной аттестации
				Всего	Лекции	Лабораторные занятия	Практические занятия			
6-05-0611-05	Компьютерная инженерия	2	4	20	12	8	-	40		зачет
		3	5	16	8	8	-		1	экзамен
	Итого			36	20	16		40	1	

План учебной дисциплины в дистанционной форме получения образования:

Код специальности	Название специальности	Курс	Семестр	Всего	Количество работ			Академ. часов на курс. работа (проект)	Форма промежуточной аттестации
					Контрольные работы	Лабораторные занятия	Индивидуальные практические работы		
6-05-0611-05	Компьютерная инженерия	2	4	216	2		1	40	зачет
		3	5	120	1		1		экзамен
	Итого			336	3		2	40	

План учебной дисциплины в заочной форме получения высшего образования, интегрированного со средним специальным образованием

Код специальности	Название специальности	Курс	Семестр	Аудиторных часов (в соответствии с учебным планом уво)				Академ. часов на курс. работу (проект)	Контрольные работы	Форма промежуточной аттестации
				Всего	Лекции	Лабораторные занятия	Практические занятия			
6-05-0611-05	Компьютерная инженерия	2	3	20	12	8/6*	-/2*	40		зачет
		2	4	16	8	8			1	экзамен
	Итого			36	20	16		40	1	

Примечание: * - только для профилизации «Компьютерные системы и сети»

Место учебной дисциплины.

Подготовка современного специалиста требует уверенного владения возможностями, предоставляемыми компьютерными технологиями. Освоение учебной дисциплины «Программирование на языках высокого уровня» обеспечивает подготовку специалиста, владеющего фундаментальными знаниями и практическими навыками в области объектно-ориентированного анализа и программирования, а также реализацией их на современных языках программирования C++ и Java. Знания и навыки, полученные при изучении учебной дисциплины, формируют базовый уровень знаний инженера-системотехника и необходимы для освоения других учебных дисциплин примерного учебного плана специальности 6-05-0611-05 «Компьютерная инженерия».

Воспитательное значение учебной дисциплины «Программирование на языках высокого уровня» заключается в формировании у обучающихся правильной культуры и стиля программирования и научного мировоззрения; развитии исследовательских умений, аналитических способностей, креативности, необходимых для решения научных и практических задач; развитии познавательных способностей и активности: творческой инициативы, самостоятельности, ответственности и организованности.

Изучение данной учебной дисциплины способствует созданию условий для формирования интеллектуально развитой личности обучающегося, которой присущи стремление к профессиональному совершенствованию, активному участию в экономической и социально-культурной жизни страны, гражданская ответственность и патриотизм.

Цели учебной дисциплины:

- приобретение теоретических знаний и практических навыков в области современных технологий объектно-ориентированной разработки программ с использованием современных стандартов;

- освоение основных принципов объектно-ориентированного проектирования и программирования программных продуктов, а также информации о современных методах и средствах программирования;

- приобретение навыков практического применения полученных знаний для решения инженерных и научно-исследовательских задач.

Задачи учебной дисциплины:

- приобретение знаний в области высокоуровневого программирования;
- освоение навыков проектирования, программирования, отладки и тестирования прикладного программного обеспечения;

- изучение принципов объектно-ориентированного программирования;
- овладение современными методами и средствами разработки прикладного программного обеспечения;

приобретение навыков разработки кроссплатформенного программного обеспечения;

освоение разнообразных возможностей, предоставляемых современными компьютерными технологиями.

В результате изучения учебной дисциплины «Программирование на языках высокого уровня» формируются следующие компетенции:

универсальные:

владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации;

быть способным к саморазвитию и совершенствованию в профессиональной деятельности;

проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности;

базовая профессиональная: применять методы и средства программирования, объектно-ориентированный подход в программировании и конструировании программ с использованием языков высокого уровня.

В результате изучения учебной дисциплины обучающийся должен:

знать:

технологии объектно-ориентированного программирования (ООП);

основные средства языка C++ для работы с объектами;

механизмы наследования, инкапсуляции и полиморфизма;

обработку исключительных ситуаций;

основы применения шаблонов и стандартные библиотеки шаблонов;

проблемы кроссплатформенности в Java;

мультипоточность и асинхронные вычисления средствами Java;

методы работы с хранилищами данных средствами Java, облачные решения;

уметь:

применять объектно-ориентированный подход при разработке программного обеспечения;

использовать потоки ввода-вывода;

разрабатывать программы с генерацией и обработкой исключительных ситуаций;

разрабатывать и использовать классы шаблонов и стандартных библиотек шаблонов;

разрабатывать прикладное программное обеспечение, используя кроссплатформенный подход;

использовать фреймворки и библиотеки в web-разработке;

владеть:

методами и приемами объектно-ориентированного подхода при проектировании и разработке прикладного программного обеспечения;

техникой создания объектно-ориентированных программных компонент и организацией их взаимодействия в программных проектах;
инструментарием разработки кроссплатформенных приложений;
основами исследовательской деятельности, навыками осуществления поиска, анализа и синтеза информации.

Перечень учебных дисциплин, усвоение которых необходимо
для изучения данной учебной дисциплины.

№ п.п.	Название учебной дисциплины	Раздел, темы
1	Основы алгоритмизации и программирования	В полном объеме
2	Арифметические и логические основы цифровых устройств	В полном объеме

1. Содержание учебной дисциплины

№ тем	Наименование разделов, тем	Содержание тем
Раздел 1. Концепция и особенности объектно-ориентированного подхода		
1.	Концептуальные основы объектно-ориентированного подхода	История создания языка C++, стандарты языка. Новые возможности языка C++. Основные направления в программировании. Парадигмы программирования.
2.	Принципы объектно-ориентированного программирования	Базовые принципы ООП. Основные положения объектной модели, ее преимущества. Абстрагирование. Модульность. Иерархия. Типизация. Пространство имен и стандартные библиотеки языка, организация ввода-вывода.
Раздел 2. Базовые абстракции объектно-ориентированного программирования		
3.	Классы и объекты	Базовые конструкции объектно-ориентированных программ: классы и объекты. Инициализация и разрушение объекта. Компоненты класса. Конструкторы и деструктор. Конструктор копирования. Семантика перемещения. Конструктор перемещения. Перегрузка и переопределение методов класса. Динамическое выделение памяти. Операторы для динамического выделения и освобождения памяти.
4.	Статические и динамические объекты	Указатели и ссылки. Умные указатели. Статические и динамические объекты. Временные объекты. Лямбда-функции.
5.	Методы и механизмы инкапсуляции	Принцип инкапсуляции. Область действия класса и доступ к компонентам класса. Спецификаторы доступа к компонентам класса. Организация внешнего доступа к локальным компонентам класса. Интерфейсные (дружественные) методы.
6.	Структурные элементы класса и методы взаимодействия объектов	Статические и константные компоненты класса. Константное выражение. Дружественные классы. Вложенные классы. Преобразование типов. Перегрузка операций.
Раздел 3. Методы и механизмы разработки объектно-ориентированных программ		
7.	Наследование в C++	Базовые и производные классы. Спецификаторы наследования. Простое и множественное наследование. Наследование членов класса и доступ к ним. Конструкторы и деструкторы при наследовании. Композиция и агрегация.
8.	Полиморфизм, механизмы его использования	Виртуальные методы класса. Виртуальные таблицы. Абстрактные классы, их назначение и свойства. Позднее и раннее связывание. Использование виртуального механизма для реализации принципа полиморфизма.
9.	Параметризация объектов в C++	Параметризованные классы и методы, их свойства. Совместное использование параметризации и принципов наследования. Параметризованные классы и статические члены. Задание значений параметров класса по умолчанию.
10.	Исключения и их обработка	Основы обработки исключений. Генерация и обработка исключений. Повторная генерация исключений. Обработка неожиданных исключений. Генерация исключений в конструкторах. Развертывание стека. Спецификация исключений. Иерархия исключений стандартной библиотеки.

№ тем	Наименование разделов, тем	Содержание тем
11.	Потоки ввода-вывода в C++. Организация работы с файлами	Понятие потока. Потоки ввода-вывода: библиотеки, классы и объекты потоков ввода-вывода. Состояние потоков ввода-вывода. Связывание классов с потоками ввода-вывода. Файловая система и ее связь с системой ввода-вывода. Текстовые, бинарные и файлы произвольного доступа. Контроль ошибок при работе с файлами.
Раздел 4. Стандартная библиотека шаблонов		
12.	Организация библиотеки STL и стандартные контейнеры	Введение в стандартную библиотеку шаблонов, основные понятия. Организация стандартной библиотеки. Типы контейнерных классов. STL-контейнеры.
13.	Итераторы, алгоритмы и распределители памяти	Категории итераторов, операции с итераторами, итераторы потоков. Стандартный распределитель памяти. Распределители памяти, определяемые пользователем. Обзор алгоритмов стандартной библиотеки.
14.	Паттерны проектирования	Назначение паттернов проектирования. Основные их виды. Решение задач проектирования с помощью паттернов.
Раздел 5. Экосистема языка Java		
15.	Положение Java в иерархии языков и фреймворков	Классификация языков программирования. Классификация фреймворков и средств разработки программного обеспечения. Положение Java в общей классификации. Экосистема языка Java. История развития языка Java. Версионность языка Java, стабильные версии языка, основные вендоры. Главные отличия языка Java. Организация работы программ на Java: JVM, JRE, JDK.
16.	Основные средства разработки в Java-экосистеме	Структура простейшей программы на Java. Сборка и запуск простейших программ на Java. Обзор IDE для разработки Java-программ. Структура типичных Java-приложений. Средства сборки Java-приложений: ant, maven, gradle. Средства запуска web-приложений: Tomcat, Jetty. Средства контроля версий кода: Git.
17.	Проблемы кроссплатформенности в Java: от JVM к web-приложениям	Виды «платформ» и проблема кроссплатформенности. Традиционный взгляд на решение проблемы кроссплатформенности: от виртуальных машин к койнтернеризации. Решение проблемы кроссплатформенности с помощью клиент-сервер архитектуры в web-приложениях. История развития Java как основного средства построения web-приложений. Web-сервисы, основные типы API в web-сервисах: SOAP, REST, GraphQL.
18.	Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке	Экосистема и набор средств Java для решения задач различной направленности. Обзор основных фреймворков и библиотек для построения web-приложений и сервисов в Java. Отличие фреймворка от библиотеки. Обзор основных принципов и средств фреймворка Spring: container, autowiring, AOP. Конфигурация Spring-приложения. Использование Spring MVC для построения web-сервисов. Вызов web-сервисов из Java приложений. Средства вызова и тестирования web-сервисов.

№ тем	Наименование разделов, тем	Содержание тем
Раздел 6. Базовые концепции языка Java		
19.	Реализация принципов ООП в Java. Синтаксис и структуры данных	Классы и объекты в Java. Реализация основных принципов ООП в Java: инкапсуляция, наследование, полиморфизм, абстракция. Типы отношений: агрегация и композиция. Основные типы данных в Java. Классы-«обёртки» над основными типами данных. Работа с коллекциями в Java. Интерфейсы и абстрактные классы. Работа со строками. Сборщик мусора в Java.
20.	Отладка и обработка ошибок средствами Java	Иерархия исключений в Java. Обработка исключений: try-catch-finally блок. Проверяемые и непроверяемые исключения. Средства логирования ошибок: Log4J, SLF4J, Logback, ELK стек. Тестирование приложений, пирамида тестирования. Автоматизированное тестирование средствами Java: unit-тестирование, интеграционное тестирование. E2E-тестирование с помощью фреймворка Selenium. TDD-подход в написании программ.
21.	Мультипоточность и асинхронные вычисления средствами Java	Мультипоточность и мультизадачность. Мультипоточность средствами Java. Решение проблемы одновременного доступа к ресурсам и потокобезопасность средствами Java: синхронизация, concurrent коллекции, пакет atomic. Организация асинхронных вызовов средствами Java.
22.	Средства функционального программирования в Java	Основные парадигмы и подходы к построению программ: императивный, процедурный, функциональный. Примеры и значение функциональных языков. Основные средства функционального подхода в Java: lambda-функции, функциональные интерфейсы, optional. Обзор Stream API и основных операторов.
Раздел 7. Основы разработки программных продуктов средствами Java		
23.	«Вечнозелёные» принципы и шаблоны разработки в Java	Основные «вечнозелёные» принципы при построении программного обеспечения. SOLID-принципы. Основные Java-паттерны для построения программ, классификация паттернов согласно GoF. Обзор основных шаблонов: стратегия, фабрика, прокси, MVC. Базовые архитектурные стили. Основы непрерывной разработки кода: Agile-методологии, CI/CD. Подходы и средства улучшения качества кода: ревью кода, тестирование, статические анализаторы.
24.	Работа с хранилищами данных средствами Java	Понятие данных. Основные типы хранилищ. Реляционные базы данных: основные принципы. Язык SQL, основные команды. Нереляционные хранилища данных: основные принципы, CAP-теорема. Виды и примеры нереляционных хранилищ: MongoDB, Redis. Средства работы с базами данных в Java: ODBC/JDBC, ORM-фреймворки, Hibernate, MyBatis. Средства работы с нереляционными хранилищами на примере MongoDB и Redis.
25.	Фронтенд-разработка в web- и мультиплатформенных приложениях	Виды фронтенда в клиент-сервер архитектуре, «толстый» и «тонкий» клиенты. Средства Java для построения десктоп приложений: Swing, AWT, JavaFx. Использование GWT/GXT для построения фронтенда в web-приложениях. Основные

№ тем	Наименование разделов, тем	Содержание тем
		технологии и фреймворки для построения клиентов web-приложений: HTML, CSS, JavaScript, Ajax, JavaScript-фреймворки. Современные принципы построения фронтенда web-приложений: адаптивная вёрстка, single-page приложения, микро-фронтенды, SEO. Мультиплатформенная разработка для мобильных устройств: гибридные приложения, кросс-платформенные приложения, PWA.
26.	Облачные решения	Основные модели облачных решений: SaaS, PaaS, IaaS. Основные поставщики облачных решений. Облачные сервисы на примере AWS. Основные подходы построения приложений в облачных сервисах: cloud native, serverless.

2. Информационно-методический раздел

2.1 Литература

2.1.1. Основная

- 2.1.1.1. Таненбаум, Э. Современные операционные системы / Э. Таненбаум, Х. Бос. – 4-е изд. – Санкт-Петербург : Питер, 2021. – 1120 с.
- 2.1.1.2. Лафоре, Р. Структуры данных и алгоритмы Java / Р. Лафоре. – 2-е изд. – Санкт-Петербург : Питер, 2018. – 704 с.
- 2.1.1.3. Васильев, А. Н. Java. Объектно-ориентированное программирование / А. Н. Васильев. – Санкт-Петербург : Питер, 2013. – 400 с.
- 2.1.1.4. Лой, М. Програмируем на Java / М. Лой, П. Нимайер, Д. Лук. – 5-е межд. изд. – Санкт-Петербург : Питер, 2023. – 544 с.
- 2.1.1.5. Луцик, Ю. А. Объектно-ориентированное программирование на языке C++ : учебное пособие / Ю. А. Луцик, В. Н. Комличенко. – Минск : БГУИР, 2008. – 266 с.
- 2.1.1.6. Рао, С. Освой самостоятельно C++ за 21 день / С. Рао. – 7-е изд. – Москва : Вильямс, 2013. – 688 с.
- 2.1.1.7. Хортон, А. Visual C++ 2010 : полный курс / А. Хортон. – Москва : Диалектика, 2009. – 1216 с.
- 2.1.1.8. Шилдт, Г. Самоучитель C++ / Г. Шилдт. – 3-е изд. – Санкт-Петербург : БХВ-Петербург, 2006. – 688 с.
- 2.1.1.9. Шилдт, Г. Искусство программирования на C++ / Г. Шилдт. – Санкт-Петербург : БХВ-Петербург, 2005. – 496 с.
- 2.1.1.10. Климова, Л. C++. Практическое программирование. Решение типовых задач / Л. Климова. – М. : Кудиц-образ, 2001. – 174 с.
- 2.1.1.11. Шилдт, Г. Java. Полное руководство / Г. Шилдт ; пер. с англ. – 10-е изд. – Санкт-Петербург : Альфа книга, 2022. – 1488 с.
- 2.1.1.12. Ричардсон, К. Микросервисы. Паттерны разработки и рефакторинга

/ К. Ричардсон. – Санкт-Петербург : Питер, 2019. – 544 с.

2.1.1.13. Страуструп, Б. Язык программирования C++ / Б. Страуструп ; пер. с англ. – 4-е изд. – Москва : Бином, 2023. – 1216 с.

2.1.1.14. Страуструп, Б. Программирование : принципы и практика использования C++ / Б. Страуструп ; пер. с англ. Д. А. Ключина. – Москва ; Санкт-Петербург : Диалектика ; Вильямс, 2015. – 1223 с.

2.1.1.15. Шилдт, Г. C++ : базовый курс / Г. Шилдт. – 3-е изд. – Москва ; Санкт-Петербург : Диалектика, 2019. – 620 с.

2.1.1.16. Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ / Г. Буч. – 3-е изд. – Москва : Вильямс, 2020.

2.1.1.17. Дейтел, Х. Как программировать на C++ / Х. Дейтел, П. Дейтел ; пер. с англ. – 8-е изд. – Москва : Бином, 2021. – 1032 с.

2.1.1.18. Java from EPAM : учебно-методическое пособие / И. Н. Блинов, В. С. Романчик. – Минск : Четыре четверти, 2020. – 560 с.

2.1.1.19. Блох, Дж. Java : эффективное программирование / Дж. Блох ; пер. с англ. – 3-е изд. – Санкт-Петербург : Диалектика, 2019. – 464 с.

2.1.1.20. Уоллс, У. Spring в действии / У. Уоллс ; пер. с англ. – 6-е изд. – Москва : ДМК Пресс, 2022. – 544 с.

2.1.1.21. Динеш, Р. Spring. Все паттерны проектирования / Р. Динеш. – Санкт-Петербург : Питер, 2019. – 320 с.

2.1.2. Дополнительная

2.1.2.1. Прата, С. Язык программирования C++. Лекции и упражнения / С. Прата. – 6-е изд. – Москва : Вильямс, 2012. – 1248 с.

2.1.2.2. Страуструп, Б. Дизайн и эволюция C++ / Б. Страуструп. – Санкт-Петербург : Питер, 2007. – 445 с.

2.1.2.3. Вайсфельд, М. Объектно-ориентированное мышление / М. Вайсфельд ; пер. с англ. – Санкт-Петербург : Питер, 2014. – 304 с.

2.1.2.4. Сьерра, К. Паттерны проектирования / К. Сьерра, Э. Фримен ; пер. с англ. – Санкт-Петербург : Питер, 2014. – 656 с.

2.1.2.5. Вербург, М. Java. Новое поколение разработки / М. Вербург, Б. Эванс ; пер. с англ. – Санкт-Петербург : Питер, 2014. – 560 с.

2.1.2.6. Хорстманн, К. Java 2 : в 2 т. Т. 1 : Основы / К. Хорстманн, Г. Корнелл ; пер. с англ. – Москва : Вильямс, 2012. – 816 с.

2.1.2.7. Хорстманн, К. Java 2 : в 2 т. Т. 1 : Основы / К. Хорстманн, Г. Корнелл ; пер. с англ. – 9-е изд. – Москва : Вильямс, 2014. – 864 с.

2.1.2.8. Хеклер, М. Spring Boot по-быстрому / М. Хеклер. – Санкт-Петербург : Питер, 2022. – 352 с.

2.2 Перечень компьютерных программ, наглядных и других пособий, методических указаний и материалов, технических средств обучения, оборудования для выполнения лабораторных работ

2.2.1. Учебно-методические пособия и материалы.

2.2.1.1. Демидович, Е. М. Конструирование программ и языки программирования (ассемблер, Си, С++) : пособие для студентов БГУИР. Ч. 1 / Е. М. Демидович. – Минск : Бестпринт, 2003. – 292 с.

2.2.1.2. Демидович, Е. М. Конструирование программ и языки программирования (ассемблер, Си, С++) : пособие для студентов БГУИР. Ч. 2 / Е. М. Демидович. – Минск : Бестпринт, 2003. – 300 с.

2.2.1.3. Конструирование программ и языки программирования [Электронный ресурс] : электронный ресурс по учебной дисциплине : 1-40 02 01. – Минск : БГУИР, 2015.

2.2.1.4. Конструирование программ и языки программирования [+ электр. вариант] : методические указания по курсовому проектированию для студентов спец. 1-40 02 01 "Вычислительные машины, системы и сети" всех форм обучения / сост.: А. В. Бушкевич, А. М. Ковальчук, И. В. Лукьянова. – Минск : БГУИР, 2010. – 30 с.

2.2.1.5. Кросс-платформенное программирование [Электронный ресурс] : электронный ресурс по учебной дисциплине : 1-40 02 01. – Минск : БГУИР, 2015.

2.2.2. Компьютерные программы.

2.2.2.1. Операционные системы Microsoft Windows (8 и выше) и Linux.

2.2.2.2. Компилятор для языка программирования С++ (рекомендуемая среда разработки MS Visual Studio 2017(или новее).

2.2.2.3. Пакет Java SE Development Kit 8.

2.2.2.4. Среда разработки IntelliJ IDEA Community Edition или Eclipse IDE for Java Developers.

2.2.2.5. Средства сборки проекта Apache Maven.

2.2.2.6. Средства тестирования Apache JMeter.

2.2.3. Технические средства обучения и оборудование.

2.2.3.1. Компьютерный класс с ПЭВМ архитектуры IBM PC с конфигурацией не хуже процессор – Intel Celeron, объем ОЗУ – 2Гб.

2.2.3.2. Проектор (монитор) для демонстрации презентаций лекций и примеров кода программ.

2.3. Перечень тем практических занятий, их название (только для профилизации «Компьютерные системы и сети»)

Целью практических занятий является закрепление теоретического курса, приобретение навыков решения задач, активизация самостоятельной работы обучающихся.

№ темы по п.1	Название практического занятия	Содержание	Обеспеченность по пункту 2.2
3 семестр			
10, 13	Генерация и обработка исключений. Проектирование контейнеров и итераторов.	Исключения. Методы обработки и генерации исключений. Разработка собственных классов обработки исключений.	2.2.1.1 - 4, 2.2.2.1 - 6, 2.2.3.1., 2.2.3.2
12, 14	Классы-контейнеры STL. Обобщенные алгоритмы.	Проектирование собственных контейнерных классов. Использование итераторов в шаблонных алгоритмах (сортировки и др.).	2.2.1.1 - 4, 2.2.2.1 - 6, 2.2.3.1., 2.2.3.2
4 семестр			
15, 16, 17	Основные средства разработки в Java-экосистеме.	Сборка и запуск простейших программ на Java. Средства сборки Java-приложений: ant, maven, gradle. Средства запуска web-приложений: Tomcat, Jetty. Конфигурация Spring-приложения.	2.2.1.5, 2.2.2.3-2.2.2.6
18, 24	Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке.	Использование Spring MVC для построения web-сервисов. Вызов web-сервисов из Java-приложений. Средства работы с базами данных в Java.	2.2.1.5, 2.2.2.3-2.2.2.6

2.4 Перечень тем лабораторных занятий, их название

Основная цель проведения лабораторных занятий состоит в закреплении теоретического материала курса, приобретении навыков выполнения эксперимента, обработки экспериментальных данных, анализа результатов, грамотного оформления отчетов.

№ темы по п.1	Наименование лабораторной работы	Содержание	Обеспеченность по пункту 2.2
3	Классы и объекты. Организация ввода-вывода. Динамическое выделение памяти	Классы и объекты. Спецификаторы доступа. Конструкторы и деструкторы. Указатели и ссылки. Динамическое выделение памяти под объекты.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
6	Дружественные функции. Перегрузка функций и операторов	Организация внешнего доступа к локальным компонентам класса. Реализация перегрузки унарных и бинарных операторов, операторов сравнения, присваивания. Перегрузка [], O(int n), new и delete.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2

№ темы по п.1	Наименование лабораторной работы	Содержание	Обеспеченность по пункту 2.2
7	Наследование. Простое и множественное наследование	Наследование классов. Разработка иерархии наследования классов. Вызов конструкторов и деструкторов при наследовании классов.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
8	Принцип полиморфизма. Виртуальные функции и абстрактные классы	Виртуальные функции. Чисто виртуальные функции. Абстрактные классы. Статический и динамический полиморфизм.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
9	Шаблоны функций и классов	Реализация шаблонных классов и функций. Наследование шаблонных классов.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
10	Генерация и обработка исключительных ситуаций	Разработка собственных классов обработки исключений. Обработчики исключительных ситуаций. Перенаправление исключительных ситуаций. Исключения стандартной библиотеки.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
11	Потоки ввода/вывода. Работа с файлами	Организация работы с текстовыми и бинарными файлами. Файлы последовательного и произвольного доступа. Перегрузка операторов ввода/вывода для работы с файлами.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
13	Классы-контейнеры, классы-итераторы, алгоритмы. STL-контейнеры	Разработка собственных классов контейнеров и классов итераторов. Алгоритмы. Применение контейнеров STL.	2.2.1.1 -4, 2.2.2.1 -6, 2.2.3.1., 2.2.3.2
16	Стандартные средства разработки в Java-экосистеме для построения web-сервисов.	Разработка простого web-приложения с использованием стандартных фреймворков и библиотек Java.	2.2.1.5, 2.2.2.3- 2.2.2.6
18	Стандартные фреймворки на примере Spring для разработки программ на Java	Разработка web-приложения средствами Java с использованием Spring фреймворка.	2.2.1.5, 2.2.2.3- 2.2.2.6
18	Конфигурация web-сервиса с помощью Spring MVC фреймворка	Разработка и конфигурация web-сервиса с использованием Spring MVC фреймворка.	2.2.1.5, 2.2.2.3- 2.2.2.6
19	Основные структуры данных в Java	Разработка средств кеширования данных с использованием стандартных структур данных Java.	2.2.1.5, 2.2.2.3- 2.2.2.6
20	Отладка и обработка ошибок средствами Java	Использование логгирования, юнит тестирования и стандартных обработчиков ошибок в web-приложении.	2.2.1.5, 2.2.2.3- 2.2.2.6
21	Мультипоточность и синхронизация доступа к ресурсам средствами Java	Использование стандартных средств синхронизации потоков для реализации конкурентной работы с ресурсами.	2.2.1.5, 2.2.2.3- 2.2.2.6

№ темы по п.1	Наименование лабораторной работы	Содержание	Обеспеченность по пункту 2.2
22	Средства функционального программирования в Java	Применение средств функционального программирования в Java для оптимизации кода.	2.2.1.5, 2.2.2.3- 2.2.2.6
24	Работа с хранилищами данных средствами Java	Использование хранилищ данных при работе с web-приложением.	2.2.1.5, 2.2.2.3- 2.2.2.6

2.5 Курсовой проект, его характеристика

Целями курсового проекта является:

овладение практическими навыками проектирования и разработки законченного, отлаженного и протестированного программного продукта с использованием методик объектно-ориентированного проектирования и языка программирования высокого уровня;

закрепление и углубление теоретических знаний, полученных при изучении курса «Основы алгоритмизации и программирования», а также данного курса;

углубление знаний по оформлению документации на программный продукт.

Курсовой проект представляет собой задачу по проектированию прикладного программного обеспечения из следующих основных типов:

задачи, связанные с расчетами;

задачи автоматизации действий;

задачи планирования;

задачи моделирования.

Курсовой проект выполняется индивидуально. По согласованию с преподавателем обучающийся может выбрать иную тему курсового проекта, отвечающую вышеперечисленным целям и требованиям.

Для курсового проекта желательна нацеленность результата на требования конкретной группы потребителей, а также реализация в программном обеспечении собственных уникальных функций и алгоритмов обработки данных.

Перечень тем курсовых проектов

1. Разработка программы вычисления по математическим формулам или методам (например, архиватор, калькулятор, построитель графиков, формирователь отчетов).

2. Разработка программы исследования и распознавания объектов или ситуаций (номеров, символов, машин, людей и т.п.).

3. Разработка менеджера организации данных (файлов, коллекций фотографий, музыки и т.п.).

4. Разработка программы просмотра или редактирования данных (текста, графики, видео и т.п.).

5. Разработка программы коммуникации данных (текста, графики, аудио- и видео-данных).

6. Разработка информационно-обучающей программы.
7. Разработка информационно-справочной программы (для различных областей и объектов хозяйства, включая задачи планирования).
8. Разработка программы моделирования (построение моделей физических, поведенческих и других процессов, включая задачи прогнозирования и искусственного интеллекта).
9. Разработка программы управления информационной системой магазина.
10. Разработка программы, организующей работу деканата.
11. Реализация справочно-информационной системы автовокзала.

2.6 Перечень рекомендуемых средств диагностики результатов учебной деятельности

Для диагностики результатов учебной деятельности могут использоваться следующие формы:

- устный опрос;
- отчеты по лабораторным работам (ЛР) с их устной защитой;
- защита контрольной работы (КР);
- защита индивидуальной практической работы (ИПР).

2.7 Контрольная работа

2.7.1 Контрольная работа для дистанционной формы получения образования

№ темы по п.1	Наименование контрольной работы	Содержание	Обеспеченность по пункту 2.2
1 – 8	КР №1. Классы и объекты. Дружественные функции и классы. Перегрузка операций. Наследование. Виртуальные функции. Абстрактные классы.	Компоненты класса. Организация внешнего доступа к компонентам класса. Дружественные методы. Переменные ссылочного типа. Перегрузка операций. Наследование, простое и множественное наследование. Использование абстрактных классов для решения прикладных программ.	2.2.1.1 - 4, 2.2.2.1 - 6, 2.2.3.1., 2.2.3.2
9,11	КР №2. Параметризованные классы. Работа с файлами. Текстовые и бинарные файлы.	Шаблоны классов и методов. Организация работы с потоками. Состояние потока. Перегрузка потоков.	2.2.1.1 - 4, 2.2.2.1 - 6, 2.2.3.1 - 2
19-23, 25	КР №3. Реализация принципов ООП в Java. Структуры данных. Обработка ошибок средствами Java. Мультипоточность. Средства функционального программирования.	Основные типы данных в Java. Классы-«обёртки» над основными типами данных. Работа с коллекциями в Java. Интерфейсы и абстрактные классы. Работа со строками. Сборщик мусора в Java. Иерархия исключений в Java. Потокбезопасность средствами Java: синхронизация, concurrent коллекции, пакет atomic. Средства функционального подхода в Java.	2.2.1.5, 2.2.2.3- 2.2.2.6

2.7.2 Контрольная работа для заочной формы получения образования, в том числе для заочной формы получения высшего образования, и интегрированного со средним специальным образованием

№ темы по п.1	Наименование контрольной работы	Содержание	Обеспеченность по пункту 2.2
16-22, 24	Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке. Структуры данных. Обработка ошибок средствами Java.	Средства сборки Java-приложений: ant, maven, gradle. Средства запуска web-приложений: Tomcat, Jetty. Конфигурация Spring-приложения. Использование Spring MVC для построения web-сервисов. Вызов web-сервисов из Java-приложений. Средства работы с базами данных в Java. Работа с коллекциями в Java. Интерфейсы и абстрактные классы. Работа со строками. Сборщик мусора в Java. Иерархия исключений в Java.	2.2.1.5, 2.2.2.3- 2.2.2.6

2.8 Индивидуальная практическая работа

Выполнение индивидуальной практической работы, предусмотренной учебным планом специальности в дистанционной форме получения образования, направлено на практическую реализацию вопросов курса при самостоятельной работе обучающегося над материалом учебной дисциплины.

№ темы по п.1	Наименование индивидуальной практической работы	Содержание	Обеспеченность по пункту 2.2
10, 12, 13, 14	ИПР №1. Генерация и обработка исключений. Проектирование контейнеров и итераторов. Классы-контейнеры STL. Обобщенные алгоритмы	Исключения. Методы обработки и генерации исключений. Разработка собственных классов обработки исключений. Проектирование собственных контейнерных классов. Использование итераторов в шаблонных алгоритмах (сортировки и др.).	2.2.1.1 - 4, 2.2.2.1 - 6, 2.2.3.1., 2.2.3.2
15-18, 24	ИПР №2. Основные средства разработки в Java-экосистеме. Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке.	Сборка и запуск простейших программ на Java. Средства сборки Java-приложений: ant, maven, gradle. Средства запуска web-приложений: Tomcat, Jetty. Конфигурация Spring-приложения. Использование Spring MVC для построения web-сервисов. Вызов web-сервисов из Java-приложений. Средства работы с базами данных в Java.	2.2.1.5, 2.2.2.3- 2.2.2.6

3. 1 Учебно-методическая карта учебной дисциплины в дневной форме получения образования

Номер раздела,	Название раздела, темы	Количество аудиторных часов			Самостоятельная работа, часы	Форма контроля знаний
		ЛК	Лаб. зан.	ПЗ		
Семестр 3						
Раздел 1. Концепция и особенности объектно-ориентированного подхода		4	-	-	16	
1	Концептуальные основы объектно-ориентированного подхода	2	-	-	8	устный опрос
2	Принципы объектно-ориентированного программирования	2	-	-	8	устный опрос
Раздел 2. Базовые абстракции объектно-ориентированного программирования		14	8/4*	-	32	
3.	Классы и объекты	4	4/4*	-	8	устный опрос, отчет по ЛР с устной защитой
4.	Статические и динамические объекты	4	-	-	8	устный опрос
5.	Методы и механизмы инкапсуляции	2	-	-	8	устный опрос
6.	Структурные элементы класса и методы взаимодействия объектов	4	4	-	8	устный опрос, отчет по ЛР с устной защитой
Раздел 3. Методы и механизмы разработки объектно-ориентированных программ		26	20/16*	-/2*	52	
7.	Наследование в С++	6	4/4*	-	12	устный опрос, отчет по ЛР с устной защитой
8.	Полиморфизм, механизмы его использования	6	4/4*	-	10	устный опрос, отчет по ЛР с устной защитой
9.	Параметризация объектов в С++	4	4/4*	-	10	устный опрос, отчет по ЛР с устной защитой
10.	Исключения и их обработка	6	4	-/2*	10	устный опрос, отчет по ЛР с устной защитой
11.	Потоки ввода-вывода в С++. Организация работы с файлами	4	4/4*	-	10	устный опрос, отчет по ЛР с устной защитой
Раздел 4. Стандартная библиотека шаблонов		8	4/4*	-/6*	32	
12.	Организация библиотеки STL и стандартные контейнеры	4	-	-/2*	12	устный опрос
13.	Итераторы, алгоритмы и распределители памяти	2	4/4*	-/2*	10	устный опрос, отчет по ЛР с устной защитой
14.	Паттерны проектирования	2	-	-/2*	10	устный опрос
Промежуточная аттестация						Зачет, курсовой проект
Итого 3 семестр		52	32/24*	-/8*	132	

Номер раздела,	Название раздела, темы	Количество аудиторных часов			Самостоятельная работа, часы	Форма контроля знаний
		ЛК	Лаб. зан.	ПЗ		
Семестр 4						
Раздел 5. Экосистема языка Java		12	12/4*		16	
15.	Положение Java в иерархии языков и фреймворков	2	-	-/ 2*	2	устный опрос
16.	Основные средства разработки в Java-экосистеме	2	4	-/ 2*	4	устный опрос, отчет по ЛР с устной защитой
17.	Проблемы кроссплатформенности в Java: от JVM к web-приложениям	4	-		4	устный опрос
18.	Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке	4	8/ 4*	-/ 2*	6	устный опрос, отчет по ЛР с устной защитой
Раздел 6. Базовые концепции языка Java		10	16/16*		24	
19.	Реализация принципов ООП в Java. Синтаксис и структуры данных	2	4/ 4*		8	устный опрос, отчет по ЛР с устной защитой
20.	Отладка и обработка ошибок средствами Java	2	4/ 4*		4	устный опрос, отчет по ЛР с устной защитой
21.	Мультипоточность и асинхронные вычисления средствами Java	2	4/ 4*		4	устный опрос, отчет по ЛР с устной защитой
22.	Средства функционального программирования в Java	4	4/ 4*		8	устный опрос, отчет по ЛР с устной защитой
Раздел 7. Основы разработки программных продуктов средствами Java		12	4/4*		14	
23.	«Вечнозелёные» принципы и шаблоны разработки в Java	2	-		4	устный опрос
24.	Работа с хранилищами данных средствами Java	4	4	-/ 2*	4	устный опрос, отчет по ЛР с устной защитой
25.	Фронтенд-разработка в web- и мультиплатформенных приложениях	4	-/ 4*		4	устный опрос
26.	Облачные решения	2	-		2	устный опрос
Промежуточная аттестация						Экзамен
Итого 4 семестр		34	32/24*	-/8*	54	
Итого по учебной дисциплине		86	64/48*	-/16*	186	

Примечание: * - только для профилизации «Компьютерные системы и сети»

3.2 Учебно-методическая карта учебной дисциплины в заочной форме получения образования, в том числе в заочной форме получения высшего образования, интегрированного со средним специальным образованием

Номер раздела	Название раздела, темы	Количество аудиторных часов			Самостоятельная работа, часы	Форма контроля знаний
		ЛК	Лаб. зан.	ПЗ		
Семестр 3¹ / 4						
Раздел 1. Концепция и особенности объектно-ориентированного подхода		-	-	-	26	
1	Концептуальные основы объектно-ориентированного подхода	-	-	-	10	устный опрос
2	Принципы объектно-ориентированного программирования	-	-	-	16	устный опрос
Раздел 2. Базовые абстракции объектно-ориентированного программирования		2	4/ -*		50	
3.	Классы и объекты	2	4/	-	20	отчет по ЛР с устной защитой
4.	Статические и динамические объекты	-	-	-	10	устный опрос
5.	Методы и механизмы инкапсуляции	-	-	-	10	устный опрос
6.	Структурные элементы класса и методы взаимодействия объектов	1	-	-	10	устный опрос
Раздел 3. Методы и механизмы разработки объектно-ориентированных программ		2	4/ 6*		74	
7.	Наследование в С++	-	-	-	16	устный опрос, защита КР
8.	Полиморфизм, механизмы его использования	-	-	-	14	устный опрос
9.	Параметризация объектов в С++	-	-	-	14	устный опрос
10.	Исключения и их обработка	2	-	-/ 2*	18	устный опрос
11.	Потоки ввода-вывода в С++. Организация работы с файлами	-	4/ 6*	-	12	отчет по ЛР с устной защитой
Раздел 4. Стандартная библиотека шаблонов		2	-	-	46	
12.	Организация библиотеки STL и стандартные контейнеры	2	-	-	16	устный опрос
13.	Итераторы, алгоритмы и распределители памяти	-	-	-	16	устный опрос
14.	Паттерны проектирования	-	-	-	14	устный опрос
Промежуточная аттестация						Зачет, курсовой проект
Итого 4 семестр		6	8/ 6*	-/ 2*	196	
Семестр 4¹ / 5						
Раздел 5. Экосистема языка Java		4	4	-	30	
15.	Положение Java в иерархии языков и фреймворков	-	-	-	4	устный опрос

Номер раздела	Название раздела, темы	Количество аудиторных часов			Самостоя тельная работа, часы	Форма контроля знаний
		ЛК	Лаб. зан.	ПЗ		
16.	Основные средства разработки в Java-экосистеме	2	-	-	6	устный опрос, защита КР
17.	Проблемы кроссплатформенности в Java: от JVM к web-приложениям	-	-	-	8	устный опрос, защита КР
18.	Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке	2	4	-	12	устный опрос, отчет по ЛР с устной защитой, защита КР
Раздел 6. Базовые концепции языка Java		2	-	-	46	
19.	Реализация принципов ООП в Java. Синтаксис и структуры данных	2	-	-	14	устный опрос, защита КР
20.	Отладка и обработка ошибок средствами Java	-	-	-	8	устный опрос, защита КР
21.	Мультипоточность и асинхронные вычисления средствами Java	-	-	-	8	устный опрос, защита КР
22.	Средства функционального программирования в Java	-	-	-	16	устный опрос, защита КР
Раздел 7. Основы разработки программных продуктов средствами Java		2	4	-	28	
23.	«Вечнозелёные» принципы и шаблоны разработки в Java	-	-	-	8	устный опрос
24.	Работа с хранилищами данных средствами Java	2	4	-/ 2*	8	устный опрос, отчет по ЛР с устной защитой, защита КР
25.	Фронтенд-разработка в web- и мультиплатформенных приложениях	-	-	-	8	устный опрос
26.	Облачные решения	-	-	-	4	устный опрос
Промежуточная аттестация						Экзамен
Итого 5 семестр		8	8	-	104	
Итого по учебной дисциплине		20	16/ 14*	-/ 2*	300	

Примечания:

1 – семестр для заочной формы получения высшего образования, интегрированного со средним специальным образованием;

* - только для профилизации «Компьютерные системы и сети»

3.4 Учебно-методическая карта учебной дисциплины в дистанционной форме получения образования

Номер раздела,	Название раздела, темы	Количество работ			Самостоятельная работа, часы	Форма контроля знаний
		КР	Лаб. зан.	ИПР		
Семестр 4						
Раздел 1. Концепция и особенности объектно-ориентированного подхода						
1	Концептуальные основы объектно-ориентированного подхода	КР №1	-	-	10	Защита КР
2	Принципы объектно-ориентированного программирования	КР №1	-	-	10	Защита КР
Раздел 2. Базовые абстракции объектно-ориентированного программирования						
3.	Классы и объекты	КР №1	-	-	10	Защита КР
4.	Статические и динамические объекты	КР №1	-	-	18	Защита КР
5.	Методы и механизмы инкапсуляции	КР №1	-	-	10	Защита КР
6.	Структурные элементы класса и методы взаимодействия объектов	КР №1	-	-	10	Защита КР
Раздел 3. Методы и механизмы разработки объектно-ориентированных программ						
7.	Наследование в С++	КР №1	-	-	22	Защита КР
8.	Полиморфизм, механизмы его использования	КР №1	-	-	12	Защита КР
9.	Параметризация объектов в С++	КР №2	-	-	24	Защита КР
10.	Исключения и их обработка	КР №2	-	ИПР №1	22	Защита КР и ИПР
11.	Потоки ввода-вывода в С++. Организация работы с файлами	КР №2	-	-	12	Защита КР
Раздел 4. Стандартная библиотека шаблонов						
12.	Организация библиотеки STL и стандартные контейнеры	-	-	ИПР №1	20	Защита ИПР
13.	Итераторы, алгоритмы и распределители памяти	-	-	ИПР №1	20	Защита ИПР
14.	Паттерны проектирования	-	-	ИПР №1	16	Защита ИПР
	Промежуточная аттестация					Зачет, курсовой проект
	Итого 4 семестр	2	-	1	216	
Семестр 5						
Раздел 5. Экосистема языка Java						
15.	Положение Java в иерархии языков и фреймворков			ИПР №2	4	Защита ИПР

Номер раздела,	Название раздела, темы	Количество работ			Самостоятельная работа, часы	Форма контроля знаний
		КР	Лаб. зан.	ИПР		
16.	Основные средства разработки в Java-экосистеме			ИПР №2	12	Защита ИПР
17.	Проблемы кроссплатформенности в Java: от JVM к web-приложениям			ИПР №2	8	Защита ИПР
18.	Фреймворки и библиотеки. Использование Spring-фреймворка в web-разработке			ИПР №2	16	Защита ИПР
Раздел 6. Базовые концепции языка Java						
19.	Реализация принципов ООП в Java. Синтаксис и структуры данных	КР №3			16	Защита КР
20.	Отладка и обработка ошибок средствами Java	КР №3			12	Защита КР
21.	Мультипоточность и асинхронные вычисления средствами Java	КР №3			12	Защита КР
22.	Средства функционального программирования в Java	КР №3			16	Защита КР
Раздел 7. Основы разработки программных продуктов средствами Java						
23.	«Вечнозелёные» принципы и шаблоны разработки в Java	КР №3			8	Защита КР
24.	Работа с хранилищами данных средствами Java			ИПР №2	8	Защита ИПР
25.	Фронтенд-разработка в web- и мультиплатформенных приложениях	КР №3			8	Защита КР
	Промежуточная аттестация					Экзамен
	Итого 5 семестр	1	1		120	
	Итого по учебной дисциплине	3	2		336	

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УО
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
С ДРУГИМИ УЧЕБНЫМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ

Перечень учебных дисциплин	Кафедра, обеспечивающая учебную дисциплину по п.1	Предложения об изменениях в содержании по изучаемой учебной дисциплине	Подпись заведующего кафедрой, обеспечивающей учебную дисциплину по п.1, с указанием номера протокола и даты заседания кафедры
1	2	3	4
Операционные системы системное программирование, Жизненный цикл разработки программного обеспечения, Функциональное программирование	Электронных вычислительных машин	Нет	<p style="text-align: center;">_____</p> <p style="text-align: center;">Б.В. Никульшин</p> <p style="text-align: center;">Протокол № 18 от 06.05.2024</p>

Заведующий кафедрой
электронных вычислительных машин

_____ Б.В. Никульшин