



Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ ДВАЖДЫ ГЕРОЯ
СОВЕТСКОГО СОЮЗА, ЛЕТЧИКА-КОСМОНАВТА А.А. ЛЕОНОВА»

«УТВЕРЖДАЮ»

И. о. проректора

_____ А.В. Троицкий

« ____ » _____ 2023 г.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И
ТЕХНОЛОГИЙ**

КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

**РАБОЧАЯ ПРОГРАММА
ДИСЦИПЛИНЫ
«ЯЗЫКИ ПРОГРАММИРОВАНИЯ»**

Направление подготовки: 27.03.04 Управление в технических системах

Профиль: Информационные системы и средства управления технологическими процессами

Уровень высшего образования: бакалавриат

Форма обучения: очная

Королев
2023


Рабочая программа является составной частью основной профессиональной образовательной программы и проходит рецензирование со стороны работодателей в составе основной профессиональной образовательной программы. Рабочая программа актуализируется и корректируется ежегодно.

Автор: Парафейников И.В. Рабочая программа дисциплины: Основы алгоритмизации и программирования – Королев МО: «Технологический университет», 2023 г.

Рецензент: к.т.н., доцент Баранова О.М.

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки бакалавров 27.03.04 Управление в технических системах и Учебного плана, утвержденного Ученым советом «Технологического университета». Протокол № 9 от 11 апреля 2023 г.

Рабочая программа рассмотрена и одобрена на заседании кафедры:

Заведующий кафедрой (ФИО, ученая степень, звание, подпись)	Баранова О.М., к.т.н., доцент 			
Год утверждения (переутверждения)	2023			
Номер и дата протокола заседания кафедры	№ 7 от 03.04.2023			

Рабочая программа согласована:

Руководитель ОПОП  :т.н., доцент Дмитренко Е.Н.

Рабочая программа рекомендована на заседании УМС:

Год утверждения (переутверждения)	2023			
Номер и дата протокола заседания УМС	№ 5 от 11.04. 2023			

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОПОП ВО

Целью изучения дисциплины является приобретение теоретических и практических навыков в области программирования на различных языках высокого уровня для решения прикладных задач в профессиональной деятельности.

Содержание дисциплины охватывает круг вопросов, связанных с формированием у бакалавров представления о современных языках программирования, к решению различных задач прикладной области с помощью ЭВМ; подготовка бакалавров к использованию современных компьютерных технологий и алгоритмов в будущей производственно-технической, проектно-конструкторской и исследовательской деятельности.

В процессе обучения студент приобретает и совершенствует следующие компетенции.

Профессиональные компетенции:

- способность проводить исследования электронных средств и электронных систем БКУ АКА (ПК-1);
- способность составлять проектно-сметную документацию на проект или программу в РКП (ПК-6).

Основными **задачами** дисциплины является:

1. Знакомство слушателей дисциплины с основными парадигмами и теоретическими основами программирования, терминологией объектно-ориентированного программирования.

2. Формирование базовых компетенций, связанных с разработкой программного обеспечения при решении профессиональных задач и представления о возможностях и особенностях объектно-ориентированных языков программирования при проектировании, разработке и отладке компьютерных программ;

3. Изучение теоретических основ программирования, методов реализации алгоритмов различного типа, базовых принципов объектно-ориентированного программирования, основных подходов машинного обучения;

4. Выработка навыков программирования и алгоритмизации с применением современных процедурных и объектно-ориентированных языков программирования;

5. Получение опыта работы с механизмами статического полиморфизма, такими как шаблоны функций и классов; поиска эффективных реализаций различных алгоритмов, практических навыков разработки прикладных программ, в том числе для методов анализа данных

После завершения освоения данной дисциплины студент должен:

Необходимые знания:

- языки программирования и языки поведенческого описания;
- аналоговую и цифровую схемотехнику,

- дисциплины естественнонаучного и математического цикла в рамках основной профессиональной образовательной программы;
- программные продукты: специализированное программное обеспечение по управлению проектами и программам, офисный пакет приложений для операционных систем;
- технические аспекты аналогичных программ организации.

Необходимые умения:

- осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, применяя современные, компьютерные и сетевые технологии;
- работать с современными системами автоматизированного проектирования и системами электронного документооборота;
- осваивать новые образцы программных, технических и информационных технологий;
- работать с информационным пространством на сервере организации для хранения, обмена и совместного использования информации по проекту или программе в РКП.

Трудовые действия:

- методами теоретических исследований электронных систем БКУ АКА;
- навыками разработки рекомендаций и заключений по использованию результатов научно-исследовательских и опытно-конструкторских работ электронных средств и электронных систем БКУ АКА;
- методами анализа чувствительности проекта или программы к изменению факторов, влияющих на параметры проекта или программы в РКП.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Языки программирования» относится к части, формируемой участниками образовательных отношений, основной образовательной программы подготовки бакалавров по направлению подготовки 27.03.04 «Управление в технических системах».

Изучение данной дисциплины базируется на ранее изученных дисциплинах: «Информатика», «Основы алгоритмизации и программирования», отдельные разделы «Электротехника» и компетенциях: УК-1, ОПК-6,9,11, ПК-2,3.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин: «Управление и информационные технологии в космических системах (профиль НИИ КС) (модуль): Системы внешнетраекторных и телеметрических измерений ракет-носителей и космических аппаратов», «Информационные технологии ракетной телеметрии (профиль 2 НПО ИТ) (модуль): Стандарты современной

телеметрии», «Информационные технологии ракетной телеметрии (профиль 2 НПО ИТ) (модуль): Теоретические основы проектирования антенн телеметрии» и при выполнении выпускной квалификационной работы бакалавра.

3. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины для студентов очной формы составляет 3 зачетных единиц, 108 часов.

Таблица 1

Виды занятий	Всего часов	Семестр 5
Общая трудоемкость	108	108
ОЧНАЯ ФОРМА ОБУЧЕНИЯ		
Аудиторные занятия	48	48
Лекции (Л)	16	16
Практические занятия (ПЗ)	32	32
Лабораторные работы (ЛР)	–	–
Самостоятельная работа	60	60
Курсовые работы (проекты)	–	–
Расчетно-графические работы	–	–
Контрольная работа, домашнее задание	+ –	+ –
Текущий контроль знаний (7 - 8, 14 - 15 недели)	Тест	+
Вид итогового контроля	Экзамен	Экзамен

4. Содержание дисциплины

4.1. Темы дисциплины и виды занятий

Таблица 2

Наименование тем	Лекции, час	Практические занятия, Час	Занятия в интерактивной форме, час	Код компетенций
Тема 1. Классификация языков программирования. Введение в язык программирования Java.	2	4	2	ПК-1, ПК-6
Тема 2. Реализация концепций объектно-ориентированного программирования в Java	2	4	2	
Тема 3. Расширенные представления объектно-ориентированного подхода.	2	4	2	
Тема 4. Создание Java-приложений	2	4	2	
Тема 5. Понятие и классификация паттернов проектирования	2	4	2	
Тема 6. Поведенческие и порождающие паттерны проектирования	2	4	2	
Тема 7. Структурные паттерны проектирования	2	4	2	
Тема 8. Паттерны наследования и паттерны объектов	2	4	2	
Итого:	16	32	16	

4.2. Содержание тем дисциплины

Тема 1. Классификация языков программирования. Введение в язык программирования Java.

Классификация языков программирования. Технология создания и выполнения программ в Java. Лексические основы языка Java. Управляющие конструкции языка Java. Ввод-вывод в Java.

Тема 2. Реализация концепций объектно-ориентированного программирования в Java.

Принципы объектно-ориентированного программирования. Создание класса Java. Тип String и операторы. Тип StringBuffer (StringBuilder). Массивы. Коллекции Java. Архивирование Java-кода.

Тема 3. Расширенные представления объектно-ориентированного подхода.

Перегружаемые методы. Сравнение объектов. Исключения.

Тема 4. Создание Java-приложений.

Интерфейсы. Вложенные классы. Тип Generic. Программирование ввода/вывода. Сериализация Java.

Тема 5. Понятие и классификация паттернов проектирования.

Определение паттерна проектирования. Связь паттернов проектирования и объектно-ориентированного программирования. Классификация паттернов проектирования.

Тема 6. Поведенческие и порождающие паттерны проектирования.

Цепочка обязанностей (Chain of responsibility). Команда (Command). Интерпретатор (Interpreter). Итератор (Iterator). Посредник (Mediator). Хранитель (Memento). Наблюдатель (Observer). Состояние (State). Стратегия (Strategy). Шаблонный метод (Template method). Посетитель (Visitor). Абстрактная фабрика (Abstract Factory). Строитель (Builder). Фабричный метод (Factory Method). Прототип (Prototype). Одиночка (Singleton).

Тема 7. Структурные паттерны проектирования.

Адаптер (Adapter). Мост (Bridge). Компонщик (Composite). Декоратор (Decorator). Фасад (Facade). Приспособленец (Flyweight). Заместитель (Proxy).

Тема 8. Паттерны наследования и паттерны объектов.

Фабричный метод (Factory Method). Интерпретатор (Interpreter). Шаблонный метод (Template Method). Адаптер (Adapter). Принципы разработки паттернов проектирования. Структура паттернов проектирования.

5. Перечень учебно-методического обеспечения для самостоятельной работы по дисциплине

«Методические указания для обучающихся по освоению дисциплины»

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Структура фонда оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведена в Приложении 1 к настоящей Рабочей программе.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Гуськова О.И. Объектно- ориентированное программирование в Java: учебное пособие / О. И. Гуськова. - Москва: МПГУ, 2019. - 240 с. - ISBN 978-5-4263-0648-6. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1020593>.

2. Васюткина, И. А. Технология разработки объектно-ориентированных программ на JAVA / Васюткина И.А. - Новосибирск: НГТУ, 2021. - 152 с.: ISBN 978-5-7782-1973-1. - Текст: электронный. - URL: <https://znanium.com/catalog/product/557111>.

3. Мархакшинов, А. Л. Практикум по программированию на языке Java: учебное пособие / А. Л. Мархакшинов. — Улан-Удэ: БГУ, 2019. — 70 с. — ISBN 978-5-9793-0016-0. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/154292>.

4. Дэвис, К. Шаблоны проектирования для облачной среды : монография / К. Дэвис ; пер. с англ. Д. А. Беликова. - Москва : ДМК Пресс, 2020. - 388 с. - ISBN 978-5-97060-807-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1094958>. – Режим доступа: по подписке

Дополнительная литература:

1. Альтман, Е. А. Платформа IntelliJ IDEA для разработки программ: учебно-методическое пособие / Е. А. Альтман, А. В. Александров, Т. В. Васеева. — Омск: ОмГУПС, 2020. — 37 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/165620>.

2. Березовская, Ю. В. Основы программирования на JAVA: лабораторный практикум: учебно-методическое пособие / Ю. В. Березовская. —Архангельск: САФУ, 2016. — 113 с. — ISBN 978-5-98450-442-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/161794>.

3. Кожомбердиева, Г. И. Программирование на языке Java: многопоточные приложения: учебное пособие / Г. И. Кожомбердиева. — Санкт-Петербург: ПГУПС, 2012. — 44 с. — ISBN 978-7641-0401-0. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/64399>.

4. Кожомбердиева, Г. И. Программирование на языке Java: создание графического интерфейса пользователя: учебное пособие / Г. И. Кожомбердиева, М. И. Гарина. — Санкт-Петербург: ПГУПС, 2012. — 67 с. — ISBN 978-5-7641-0402-7. — Текст: электронный // Лань : электронно библиотечная система. — URL: <https://e.lanbook.com/book/64398>.

5. Никитенкова, С. П. Многопоточное программирование на языке JAVA: учебно-методическое пособие / С. П. Никитенкова. — Нижний Новгород: ННГУ им. Н. И. Лобачевского, 2015. — 90 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/144990>.

6. Программирование на языке Java: работа со строками и массивами. Методические указания: методические указания. — Санкт-Петербург:

ПГУПС, 2015. — 24 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL:

<https://e.lanbook.com/book/66410>.

7. Бабушкина И. А. Практикум по объектно-ориентированному программированию: учебное пособие / И. А. Бабушкина, С. М. Окулов. — 5-е изд. — Москва: Лаборатория знаний, 2020. — 369 с. — ISBN 978-5-00101-780-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/135561>

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <https://biblioclub.ru/> - Университетская библиотека ONLINE.
2. <https://www.java.com/ru/> - Сайт разработчика Java
3. <https://www.rucont.ru/> - Электронная библиотека Руконт.
4. <https://znanium.com/> - Электронная библиотека издательства Инфра-М.
5. <https://e.lanbook.com/> - Электронная библиотека Лань.
6. <https://www.jetbrains.com/ru-ru/idea/> - Сайт разработчика IntelliJ IDEA.
7. <https://github.com/> - Веб-сервис для хостинга IT-проектов и их совместной разработки.
8. <https://www.eclipse.org/> - Сайт разработчика Eclipse
9. <https://jug.ru/> - Сайт Java-разработчиков <https://code.visualstudio.com/>

9. Методические указания для обучающихся по освоению дисциплины

Методические указания для обучающихся по освоению дисциплины приведены в Приложении 2 к настоящей программе.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения:

Eclipse IDE, IntelliJ IDEA, JDK Java SE8 и выше, Notepad++.

Информационные справочные системы:

1. Электронные ресурсы образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Языки программирования»

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционные занятия:

- аудитория, оснащенная презентационной техникой (проектор, экран);
- комплект электронных презентаций/слайдов.

Практические занятия:

- учебный класс, оснащенный вычислительной техникой (ПК) и доступом к Интернет-ресурсам.
- рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
- рабочие места студентов, оснащенные компьютерами с доступом в Интернет.

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И
ТЕХНОЛОГИЙ
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО
ДИСЦИПЛИНЕ**

«ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

(Приложение 1 к рабочей программе)

Направление подготовки: 27.03.04 Управление в технических системах

Профиль: Информационные системы и средства управления технологическими процессами

Уровень высшего образования: бакалавриат

Форма обучения: очная

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п/п	Индекс компетенции	Содержание компетенции (или ее части)*	Раздел дисциплины, обеспечивающий формирование компетенции (или ее части)	В результате изучения раздела дисциплины, обеспечивающего формирование компетенции (или ее части), обучающийся должен:		
				Необходимые знания	Необходимые умения	Трудовые действия
	ОПК-1	Способен анализировать задачи профессиональной деятельности на основе положений, законов и методов в области естественных наук и математики.	Тема 1. Классификация языков программирования. Введение в язык программирования Java. Тема 2. Реализация концепций объектно-ориентированного программирования в Java Тема 3. Расширенное представление объектно-ориентированного подхода	Знает положения, законы и методы в области естественных наук и математики	Умеет использовать положения, законы и методы в области естественных наук и математики для анализа задач профессиональной деятельности	Имеет навыки анализа задач профессиональной деятельности и на основе положений, законов и методов в области естественных наук и математик
1	ОПК-6	Способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и	Тема 4. Создание Java-приложений Тема 5. Понятие и классификация паттернов проектирования Тема 6.	алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и	разрабатывать и использовать алгоритмы и программы, современные информационные технологии, методы и средства контроля,	практически и опытом разработки и использования алгоритмов и программ, современных информационных технологий,

		управления, пригодные для практического применения в сфере своей профессиональной деятельности	Поведенческие и порождающие паттерны проектирования Тема 7. Структурные паттерны проектирования Тема 8. Паттерны наследования и паттерны объектов	управления	диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности	методов и средств контроля, диагностики и управления, пригодные в сфере своей профессиональной деятельности
--	--	--	---	------------	--	---

2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

<i>Код компетенции</i>	<i>Инструмент, оценивающий сформированность компетенции</i>	<i>Показатель оценивания компетенции</i>	<i>Критерии оценки</i>
ОПК-1 ОПК-6	Письменное задание	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 90% правильных ответов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> • компетенция освоена на продвинутом уровне – 70% правильных ответов; • компетенция освоена на базовом уровне – от 51% правильных ответов; <p>В) не сформирована (компетенция не сформирована) – менее 50% правильных ответов</p>	<p>1. Проводится в форме письменной работы</p> <p>2.Время, отведенное на процедуру – 10 - 15 мин.</p> <p>Неявка – 0.</p> <p>Критерии оценки:</p> <p>1.Соответствие ответа заявленной тематике (0-5 баллов).</p> <p>Максимальная сумма баллов - 5 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля. Оценка проставляется в электронный журнал.</p>

	<p>Практическое задание</p>	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 90% правильных ответов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> • компетенция освоена на продвинутом уровне – 70% правильных ответов; • компетенция освоена на базовом уровне – от 51% правильных ответов; <p>В) не сформирована (компетенция не сформирована) – менее 50% правильных ответов</p>	<p>1. Проводится в форме практического задания</p> <p>2.Время, отведенное на процедуру – 20-30 мин.</p> <p>Неявка – 0.</p> <p>Критерии оценки:</p> <p>1.Соответствие ответа заявленной тематике (0-5 баллов).</p> <p>Максимальная сумма баллов - 5 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля. Оценка проставляется в электронный журнал.</p>
	<p>Контрольная работа</p>	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 90% правильных ответов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> • компетенция освоена на продвинутом уровне – 70% правильных ответов; • компетенция освоена на базовом уровне – от 51% правильных ответов; <p>В) не сформирована (компетенция не сформирована) – менее 50% правильных ответов</p>	<p>1. Проводится в форме письменной работы</p> <p>2.Время, отведенное на процедуру – семестр.</p> <p>Неявка на защиту контрольной работы – 0.</p> <p>Критерии оценки:</p> <p>1.Соответствие содержания контрольной работы заявленной тематике (1 балл).</p> <p>2.Качество источников и их количество при подготовке работы (1 балл).</p> <p>3.Владение информацией и способность отвечать на вопросы аудитории (1 балл).</p>

			<p>4. Качество самой представленной работы (1 балл).</p> <p>5. Использование специализированного программного обеспечения (1 балл).</p> <p>6. Оригинальность подхода и всестороннее раскрытие выбранной тематики (1 балл).</p> <p>Максимальная сумма баллов - 6 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля.</p> <p>Оценка проставляется в электронный журнал.</p>
--	--	--	---

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Примерная тематика практических заданий:

1. Реализовать паттерн Абстрактная фабрика.
2. Реализовать паттерн Строитель.
3. Реализовать паттерн Фабричный метод.
4. Реализовать паттерн Наблюдатель.
5. Реализовать паттерн Цепочка обязанностей.
6. Реализовать паттерн Посетитель.
7. Реализовать паттерн Стратегия.
8. Реализовать паттерн Адаптер.
9. Реализовать паттерн Мост.

Примерная тематика заданий на контрольную работу:

Составить программу решения задачи с использованием файла данных и выполнить ее на ПК.

Вариант	Условие задачи
1	Реализовать интерфейс для создания множества связанных между собой или независимых объектов, конкретные классы которых неизвестны.
2	Преобразовать существующий интерфейс класса в другой интерфейс, который понятен клиентам. При этом обеспечить совместную работу классов, невозможную без данного паттерна из-за несовместимости интерфейсов.
3	Отделить абстракцию класса от его реализации, чтобы появилась возможность независимо изменять то и другое.
4	Отделить создание сложного объекта от его представления, позволяя использовать один и тот же процесс разработки для создания различных представлений.
5	Инкапсулировать запрос в виде объекта, обеспечивая параметризацию клиентов типом запроса, установление очередности запросов, протоколирование запросов и отмену выполнения операций.
6	Сгруппировать объекты в иерархические структуры для представления отношений типа "часть-целое", чтобы позволить клиентам работать с единичными объектами так же, как с группами объектов.
7	Специфицировать зависимость типа "один ко многим" между различными объектами, так что при изменении состояния одного объекта все зависящие от него получают извещение и автоматически обновляются.
8	Подменить выбранный объект другим объектом для управления контроля доступа к исходному объекту.
9	Для выбранного класса обеспечить выполнение требования единственности экземпляра и предоставления к нему полного доступа.
10	Позволить выбранному объекту варьировать свое поведение при изменении внутреннего состояния. При этом должно создаваться впечатление, что изменился класс объекта.
11	Определить множество алгоритмов, инкапсулируя их все и позволяя подставлять один вместо другого. При этом позволить изменять алгоритм независимо от клиента, который им пользуется.
12	Применить несколько фабричных методов для создания целого семейства или группы взаимосвязанных объектов.
13	Реализовать создание новых объектов с помощью прототипов.
14	Определить интерфейс для создания одного объекта, но предоставить подклассам решать, какой класс создать. Позволить классу отложить создание экземпляров для подклассов.
15	Реализовать тактику задержки создания объекта, вычисления значения или какого-либо другого дорогостоящего процесса до тех пор, пока он не понадобится в первый раз.
16	Составить объекты в древовидные структуры для представления

	иерархий части-целого.
17	Добавить функциональность в иерархию классов без изменения иерархии.
18	Реализовать пустой интерфейс для связи метаданных с классом.
19	Объединить несколько связанных элементов, таких как классы, синглтоны, методы, используемые глобально, в единую концептуальную сущность.
20	Реализовать способ последовательного доступа к элементам агрегатного объекта без раскрытия его базового представления.

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

<i>Неделя текущей/промежуточно го контроля</i>	<i>Вид оценочного средства</i>	<i>Код компетенций, оценивающих знания, умения, навыки</i>	<i>Содержание оценочного средства</i>	<i>Требования к выполнению</i>	<i>Срок сдачи (неделя семестра)</i>	<i>Критерии оценки по содержанию и качеству с указанием баллов</i>
Согласно графику учебного процесса	Тестирование	ОПК-1 ОПК-6	25 вопросов	Компьютерное тестирование. Время отведенное на процедуру – 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка – 0 Удовлетворительно – от 51% правильных ответов. Хорошо – от 70% правильных ответов. Отлично – от 90% правильных ответов.
Согласно графику учебного процесса	Тестирование	ОПК-1 ОПК-6	25 вопросов	Компьютерное тестирование. Время отведенное на процедуру – 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка -0 Удовлетворительно - от 51% правильных ответов. Хорошо - от 70%. Отлично – от 90%.
Согласно графику учебного процесса	зачет	ОПК-1 ОПК-6	2 вопроса	Зачет проводится в письменной форме, путем ответа на вопросы. Время отведенное на процедуру	Результаты предоставляются в день проведения зачета	Критерии оценки: «Зачтено»: – знание основных понятий предмета; – умение использовать и применять полученные знания на практике; – работа на семинарских занятиях; – знание основных научных теорий, изучаемых предметов;

				ру – 30 минут		<ul style="list-style-type: none"> – ответ на вопросы билета. «Не зачтено»: – демонстрирует частичные знания по темам дисциплин; – незнание основных понятий предмета; – неумение использовать и применять полученные знания на практике; – не работал на семинарских занятиях; – не отвечает на вопросы.
Согласно графику учебного процесса	Экзамен	ОПК-1 ОПК-6	2 вопроса, 1 практическое задание	<p>Экзамен проводится в 2 этапа: устной форме, путем ответа на вопросы и демонстрация результатов выполнения практического задания.</p> <p>Время отведенное на процедуру – 30 минут.</p>	Результаты предоставляются в день проведения экзамена	<p>Критерии оценки:</p> <p>«Отлично»:</p> <ul style="list-style-type: none"> • знание основных понятий предмета; • умение использовать и применять полученные знания на практике; • работа на практических занятиях; • знание основных научных теорий, изучаемых предметов; • ответ на вопросы билета. <p>«Хорошо»:</p> <ul style="list-style-type: none"> • знание основных понятий предмета; • умение использовать и применять полученные знания на практике; • работа на практических занятиях; • знание основных научных теорий, изучаемых предметов; • ответы на вопросы билета неполные <p>«Удовлетворительно»:</p> <ul style="list-style-type: none"> • демонстрирует частичные знания по темам дисциплин; • незнание неумение использовать и применять полученные знания на практике; • не работал на практических занятиях; <p>«Неудовлетворительно»:</p> <ul style="list-style-type: none"> • демонстрирует частичные знания по темам дисциплин; • незнание основных понятий предмета; • неумение использовать и применять полученные

						знания на практике; • не работал на практических занятиях; не отвечает на вопросы.
--	--	--	--	--	--	---

Итоговое начисление баллов по дисциплине осуществляется в соответствии с разработанной и внедренной балльно-рейтинговой системой контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся.

Примерная тематика тестовых заданий:

1. Объектно-ориентированный подход к проектированию программных продуктов основан на следующих принципах:

- 1) выделение классов объектов;
- 2) установление характерных свойств объектов и методов их обработки
- 3) создание иерархии классов, наследовании свойств объектов и методов их обработки
- 4) Ничего из выше перечисленного

2. Структурный подход использует:

- 1) диаграммы потоков данных (информационно-технологические схемы) – показывают процессы и информационные потоки между ними с учетом "событий", инициирующих процессы обработки;
- 2) диаграммы декомпозиции – структура и декомпозиция целей, функций управления, приложений
- 3) структурные схемы – архитектура программного продукта в виде иерархии взаимосвязанных программных модулей с идентификацией связей между ними, детальная логика обработки данных программных модулей (блок-схемы).
- 4) Ничего из выше перечисленного.

3. Прародителями всех языков ООП является

- 1) Симула
- 2) Delphi
- 3) C#
- 4) Объектом объектно-ориентированного программирования называется ...

4. Совокупность переменных состояния и связанных с ними методов (операций)

- 1) группа данных и методов (функций) для работы с этими данными
- 2) функция или процедура, выполняющие определенные действия
- 3) характеристика, назначенная элементу класса

5. Выберите наиболее подходящее определение Класса.

- 1) Тип, содержащий набор функций
- 2) Тип, который отображает состояние некоторого объекта
- 3) Тип, описывающий поведение некоторой сущности
- 4) Тип, описывающий характеристики и поведение объекта

6. Двумерным называется массив, элементы которого расположены в виде...

- 1) множества измерений

- 2) квадратной таблицы
- 3) прямоугольной таблицы

7. Строка Readln (a[i, j])

- 1) ожидает ввод размера массива
- 2) ожидает ввод элемента массива
- 3) ожидает вывод массива

8. Выберите язык ООП

- 1) Borland Delphi
- 2) Pascal
- 3) Basic

9. Любое условие в программном коде записывается...

- 1) IF...THEN...ELSE
- 2) Begin...END
- 3) FOR..TO..DO

10. В каком разделе задаются константы?

- 1) В var
- 2) после begin
- 3) перед var

11. Как записывается цикл в программном коде?

- 1) IF...THEN...ELSE
- 2) Begin...END
- 3) FOR..TO..DO

12. Выберите правильную форму записи в var?

- 1) имя_переменной:тип;
- 2) тип:имя_переменной;

13. Для расширения возможностей можно.

- 1) Подключить дополнительные библиотеки
- 2) Создать дополнительный цикл

14. Любой язык программирования переводит свой код в

- 1) машинный язык
- 2) Java Script
- 3) Остается неизменным

15. Какие языки программирования набирают популярность

- 1) ООП
- 2) Консольные

16. Можно ли сказать, что Delphi является универсальным языком программирования?

- 1) Да
- 2) Нет
- 3) Это не язык программирования

17. Для исправления ошибок в исходном коде, во время компиляции используется

- 1) декомпилятор
- 2) дебаггер

18. Для правильного сохранения проекта следует

- 1) Файл-Сохранить

- 2) Файл-Сохранить Как
- 3) Файл-Сохранить проект Как

19. Для чего предназначена технология программирования?

- 1) Для решения вычислительных обязанностей и финансовых служб
- 2) Для обработки точного изображения и звука
- 3) Для обработки больших структурированных объемов информации
- 4) Для создания инструментальных программных инструментов информационных технологий
- 5) Для обработки текстовой информации

20. Укажите, какое расширение имеет файл проекта

- 1) dpr
- 2) dcu
- 3) pas
- 4) exe
- 5) dfm

21. Для изменения значений свойств компонентов в процессе разработки программы используется:

- 1) Property Editor
- 2) Image Editor
- 3) Object Inspector
- 4) Code Editor
- 5) Project Options

22. Основные принципы ООП -

- 1) Инкапсуляция, наследование, полиморфизм
- 2) наследование, дедукция, инкапсуляция
- 3) Полиграфия, делегирование, наследование
- 4) Индукция, наследование, полиморфизм
- 5) Инкапсуляция, наследование, делегирование

23. Выберите последовательность областей видимости, упорядоченную по возрастанию видимости методов.

- 1) Protected, public, private
- 2) Public, protected, private
- 3) Private, protected, public
- 4) Protected, public, private
- 5) Public, private, protected

24. Выберите выражение, в результате которого получено значение $c=3$, если $a=14$ и $b=4$

- 1) $c:=b \text{ mod } a$
- 2) $c:= a \text{ mod } b$
- 3) $c:=a/b$
- 4) $c:=b \text{ div } a$
- 5) $c:= a \text{ div } b$

25. Укажите компонент Контейнер

- 1) PageControl
- 2) StatusBar
- 3) Memo

- 4) Panel
- 5) Edit

26. Возможность использовать одинаковые имена для методов, входящих различные классы называется...

- 1) Наследование
- 2) Метоморфизм
- 3) Декапсуляция
- 4) Инкапсуляция
- 5) Полиморфизм

27. Конструкторы...

- 1) предназначены для создания таблицы виртуальных методов
- 2) являются статическими методами
- 3) служат для возможности наследования виртуальных методов или их переопределения
- 4) они присутствуют во всех объектных типах (классах) и предназначены для создания экземпляров объектов
- 5) выполняются обязательно перед первым использованием виртуальных методов

28. Укажите сколько файлов будет создано при сохранении проекта, в состав которого входит только одна форма

- 1) 7
- 2) 6
- 3) 4
- 4) 5
- 5) 2

29. Массивы бывают

- 1) Числовые и вещественные
- 2) Одномерные и многомерные
- 3) Одномерные и разномерные
- 4) Строчные и числовые
- 5) Символьные или файловые

30. Основные операции с массивами - это

- 1) Перебор элементов массива, нахождение номера индекса
- 2) Подсчет индекса, количества элементов
- 3) Суммирование индексов, ввод значений в элементы массива
- 4) Определение типа элемента массива, сравнение элементов массива
- 5) Ввод и вывод значений в элементы массива, поиск максимального или минимального элемента, нахождение суммы элементов массива

31. Укажите различие функций от процедуры:

- 1) В теме функций указываются списки формальных параметров
- 2) Функция возвращает значение, а при вызове функций в программе, используется как переменная в выражении
- 3) В функции описываются все метки, константы, типы и переменные
- 4) Нет различия
- 5) Можно вызвать функцию в любом разделе главной программы

32. Что такое алгоритм

- 1) Алгоритм - это совокупность требования к программе
- 2) Алгоритм - это индивидуальные варианты исполнителю для выполнения какого-либо действия.
- 3) Алгоритм - это порядок команд для ЭВМ.
- 4) Алгоритм - это совокупность справочников и правил к исполнителю.
- 5) Алгоритм - это точное и понятное предписание, определяющее процесс перехода от исходных данных к результату.

33. Формальные параметры делятся на ...

- 1) Значения, переменные, константы
- 2) Константы, локальные
- 3) Локальные, глобальные
- 4) Переменные, глобальные
- 5) Фактические, глобальные

34. Блок-схема это

- 1) Способ представления алгоритма
- 2) Вид алгоритма
- 3) Решение задачи
- 4) Программа
- 5) Команда алгоритма

35. Массив - это

- 1) нефиксированная структура данных, содержащих переменные одного типа данных;
- 2) нефиксированная структура данных, содержащих переменные разного типа данных;
- 3) фиксированный набор данных, имеющий общее имя, содержащий переменные разного типа данных
- 4) структура данных, содержащих переменные одинакового типа и имеющих одинаковые значения.
- 5) упорядоченная последовательность однотипных данных, расположенных на носителе информации

36. Последовательность символов представляет собой

- 1) пользовательский тип данных
- 2) простой тип данных
- 3) символьный тип данных
- 4) скалярный тип данных
- 5) строковый тип данных

37. Слова, смысл и способ употребления которых задан раз и навсегда называют:

- 1) Серией
- 2) Составными словами
- 3) Командами
- 4) Служебными словами
- 5) Простыми словами

38. Свойство алгоритма «дискретность» означает:

- 1) являются слова или тексты
- 2) Способность алгоритма давать правильные результаты решения задач

- 3) Предлагаемые действия должны быть понятными и единственно возможными
- 4) Пригодность алгоритма для решения однотипных задач
- 5) Алгоритм состоит из отдельных шагов. Каждый шаг алгоритма – это некоторое законченное действие.

39. Линейным называется алгоритм, при котором...

- 1) последовательность действий совершается одна за другой в порядке их следования;
- 2) Используется оператор цикла
- 3) Используется оператор безусловного перехода
- 4) Используется оператор FOR
- 5) Используется оператор GOTO

40. Выберите идентификатор, который используется для обозначения символьного типа данных

- 1) varchar
- 2) string
- 3) symbol
- 4) char
- 5) comp

3.2 Примерная тематика контрольной работы, выполняемой с применением программных средств специального и общего назначения на компьютерах:

1. Реализовать приложение, вычисляющее для последовательности чисел, представляющих величины углов, следующие тригонометрические функции:

- сумма синусов/косинусов значений,
- разность синусов/косинусов значений,
- произведение синусов/косинусов введенных значений.

Единица измерения углов должна указываться как параметр командной строки. Для представления единиц измерения и используйте константы с типом int. Для хранения значений, необходимо использовать массив с типом double.

Объявления методов должны выглядеть следующим образом:

```
/**
```

```
@param angles – массив значений углов;
```

```
@param type -единица измерения, имеющее значение одной из констант
```

```
@return сумму синусов значений углов;
```

```
*/
```

```
public static double
```

```
getSumSinuses(double[] angles, int type);
```

2. С помощью перегрузки реализуйте методы, позволяющие указывать количество значащих знаков для результатов выполнения операций.

3. Дополнить приложение, таким образом, чтобы имелась возможность формирования исходной числовой последовательности числами, сформированными генератором случайных чисел;

4. Реализовать класс объявляемой исключительной ситуации, для проверки корректности вводимых данных. Объект данной исключительной ситуации

содержать в себе информацию о значении, обработка которого вызвала данную исключительную ситуацию.

5. Реализовать класс, в котором будут содержаться все описанные выше функциональные возможности.

Типовые вопросы, выносимые на экзамен

1. Эволюция методологий программирования. Парадигмы программирования.
2. Основные принципы объектного подхода. Абстрагирование.
3. Основные принципы объектного подхода. Инкапсуляция.
4. Основные принципы объектного подхода. Модульность.
5. Основные принципы объектного подхода. Иерархия.
6. Основные принципы объектного подхода. Типизация.
7. Основные принципы объектного подхода. Параллелизм. Сохраняемость.
8. Объект с точки зрения ООП. Состояние. Поведение.
9. Объект с точки зрения ООП. Идентичность и жизненный цикл объектов.
10. Объект с точки зрения ООП. Взаимоотношения между объектами.
11. Классы. Природа классов. Мета модель. Инстанцирование.
12. Классы. Структура класса. Абстрактные классы и интерфейсы.
13. Классы. Отношения между классами. Ассоциация и агрегация.
14. Классы. Иерархии классов. Зависимость.
15. Модель памяти и структура программы. Классы памяти. Ссылки.
16. Перегрузка операторов.
17. Жизненный цикл объекта. Инициализация массивов. Конструкторы и деструкторы. Порядок вызова конструкторов и деструкторов при наследовании.
18. Варианты реализации отношения клиент-сервер. Объекты при передаче параметров и возврате из методов.
19. Виртуальная машина. Структура программ. Типы переменных в Java. Принципы работы ClassLoader.
20. Средства абстракции Java. Структура класса. Статические члены.
21. Внутренние и вложенные классы. Статический и динамический контекст. Локальные и анонимные классы.
22. Средства инкапсуляции Java. Поддержка модульности. Пакеты.
23. Представление иерархических отношений. Наследование. Интерфейсы и абстрактные классы.
24. Агрегация и зависимость от времени жизни. Реализация отношений клиент-сервер. Стандартные контейнеры.
25. Типизация. Правила преобразования типов. instanceof и ClassCastException. Класс Class. Родовые компоненты (Generics)
26. Средства поддержки параллелизма. Активные и пассивные объекты. Класс Object.
27. Использование Thread и Runnable. Пул потоков, назначение и принципы реализации.

- 28.Исключения. Обработка исключительных ситуаций. Блок finally
- 29.Сохраняемость. Serializable и Externalizable. Программирование распределенных приложений. Java RMI
- 30.Модель безопасности Java. Policy, Permissions, AccessController.
- 31.Графическая подсистема. Основы AWT, Applet, Swing components. Событийная модель.
- 32.Средства поддержки Java машины. System, Runtime, сборка мусора. Ввод вывод, пакет java.io

Методические указания для обучающихся по освоению дисциплины

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И
ТЕХНОЛОГИЙ
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО
ОСВОЕНИЮ ДИСЦИПЛИНЫ
«ЯЗЫКИ ПРОГРАММИРОВАНИЯ»**

(Приложение 2 к рабочей программе)

Направление подготовки: 27.03.04 Управление в технических системах

**Профиль: Информационные системы и средства управления
технологическими процессами**

Уровень высшего образования: бакалавриат

Форма обучения: очная

1. Общие положения

Целью дисциплины является формирование у обучающихся системного представления об особенностях создания программ с помощью объектно-ориентированного подхода и формирования практических навыков по созданию, отладке и модификации прикладных программ, написанных в объектно-ориентированном стиле.

Основными **задачами** дисциплины являются: изучение средств объектно-ориентированного программирования языка Java, платформы Java, стандартной библиотеки классов, основ многопоточного и распределенного программирования, безопасности программных систем, использующих технологию Java.

2. Указания по проведению практических занятий

Практическая работа 1.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 1. Основы языка объектно-ориентированного программирования Java.

- Лексические основы языка Java.
- Управляющие конструкции языка Java.
- Ввод-вывод в Java.

Продолжительность занятия 4 часа.

Практическая работа 2.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 2. Реализация концепций объектно-ориентированного программирования в Java.

- Создание класса Java
- Тип String и операторы
- Тип StringBuffer (StringBuilder)

Продолжительность занятия 4 часа.

Практическая работа 3.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 2. Реализация концепций объектно-ориентированного программирования в Java.

- Массивы
- Коллекции Java
- Архивирование Java-кода

Продолжительность занятия 4 часа

Практическая работа 4.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 3. Расширенные представления объектно-ориентированного подхода.

- Перегружаемые методы

Продолжительность занятия 4 часа.

Практическая работа 5.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 3. Расширенные представления объектно-ориентированного подхода.

- Сравнение объектов.

- Исключения.

Продолжительность занятия 4 часа.

Практическая работа 6.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Создание Java-приложений.

- Интерфейсы.

- Программирование ввода/вывода

Продолжительность занятия 4 часа.

Практическая работа 7.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Создание Java-приложений.

- Вложенные классы

- Тип Generic

Продолжительность занятия 4 часа.

Практическая работа 8.

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Создание Java-приложений.

- Сериализация Java

Продолжительность занятия 4 часа.

3. Указания по проведению лабораторного практикума

Не предусмотрено учебным планом.

4. Указания по проведению самостоятельной работы студентов

№ п/п	Наименование блока (раздела) дисциплины	Виды СРС
1	Тема 1. Основы алгоритмизации	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <ol style="list-style-type: none"> Составить алгоритм и написать программу вычисления значения функции по заданному значению переменных: $e^x - \frac{y^2 + 12xy - 3x^2}{18y - 1}$ $2ctg(3x) - \frac{\ln \cos x}{\ln(1 + x^2)}$ Записать арифметическое выражение на алгоритмическом языке и указать порядок выполнения операций: $D = \frac{\sqrt{x + b} - a + \ln y}{arctg(b + a)}$ Составить алгоритм и написать программу, реализующую решение задачи: дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба. Составить программу вычисления объема цилиндра и конуса, которые имеют одинаковую высоту H и одинаковый радиус основания R.
2	Тема 2. Основы языков программирования	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <ol style="list-style-type: none"> Составить структурную схему алгоритма и проект программы для решения задачи: (исходные данные ввести с клавиатуры, результат вывести на экран) четырехугольник задан координатами своих вершин. Найти его периметр. Вершины ввести с клавиатуры. Написать программу определения катета и площади прямоугольного треугольника, по заданным катету и гипотенузе. Длины катета и гипотенузы ввести с экрана. Написать программу для решения задачи нахождения корней квадратного уравнения, введя с экрана коэффициенты a, b, c (коэффициент a не равен 0), для которых дискриминант положителен
3	Тема 3. Выражения и операторы. Структурное программирование	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <ol style="list-style-type: none"> Написать программу, которая по введенному номеру

	е	<p>единицы измерения (1 – дециметр, 2 – километр, 3 – метр, 4 – миллиметр, 5 – сантиметр) или длине отрезка L выдавала бы соответствующее значение длины отрезка в метрах.</p> <p>2. Составить блок-схему и написать программу, которая при вводе числа в диапазоне от 1 до 99 добавляет к нему слово "копейка" в правильной форме. Например, 1 копейка, 5 копеек, 42 копейки.</p> <p>3. Написать блок-схему и программу для решения задачи: дано трехзначное число. Определить: а) верно ли, что все его цифры одинаковые? б) есть ли среди его цифр одинаковые?</p> <p>4. Составить блок-схему и написать программу с использованием оператора выбора варианта для решения задачи: дано целое число k ($1 < k < 365$). Определить, каким днем недели (понедельник, вторник, воскресенье) является k-день невысокосного года, если 1 января – понедельник.</p>
4	Тема 4. Массивы	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <p>1. Составить блок-схему и написать программу для решения задачи: дана действительная матрица размером $n \times m$. Требуется преобразовать матрицу: поэлементно вычесть последнюю строку из всех строк, кроме последней.</p> <p>2. Составить программу, которая заполняет квадратную матрицу порядка n натуральными числами 1, 2, 3, ... ,n^2, записывая их «по спирали».</p> <p>3. Написать блок-схему и программу для решения задачи: все массивы в заданиях объявить как динамические. Ввести массив из 11 вещественных чисел. Создать новый массив из элементов исходного, не превышающих среднее арифметическое массива. Определить наименьший положительный элемент нового массива.</p>
5	Тема 5. Циклические конструкции	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <p>1. Написать блок-схему и составить программу, которая запрашивает пароль (четырёхзначное число) до тех пор, пока количество неверных ответов не превысит три.</p> <p>2. Составить программу перевода данного натурального числа из десятичной системы счисления в двоичную.</p> <p>3. Написать программу поиска произведения последовательности чисел, вводимых с клавиатуры, предшествующих первому введенному нулю. Контрольный пример: 1,2,3,-4,5,-2,0.</p>
6	Тема 6. Строки	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <p>1. Написать программу, которая во введенной с клавиатуры строке преобразует буквы русского алфавита в латинские.</p> <p>2. Написать программу, которая во введенной с клавиатуры</p>

		<p>строке меняет четные слова с нечетными.</p> <p>3. Напишите программу, в которой с клавиатуры вводятся две строки символов. К строке с наибольшей длиной добавить текст, содержащийся в другой строке.</p> <p>4. Напишите программу, в которой с клавиатуры вводятся две строки символов. К строке с наименьшей длиной добавить текст, содержащийся в другой строке.</p>
7	Тема 7. Функции	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <p>1. Составить схему алгоритма и написать программу вычисления таблицы значений функции $f(x)$ на промежутке от -2 до 3 с шагом 0,5:</p> $f(x) = \sum_{k=1}^5 \frac{(-2)^{k+1}(k+1)!}{x+2.5)^{k+1}}$ <p>2. Составить схему алгоритма и написать программу вычисления таблицы значений функции $f(x)$ на промежутке от -2 до 2 с шагом 0,4:</p> $f(x) = \begin{cases} \sqrt[5]{x^3}, & \text{если } x > 0 \\ x^2 + \sum_{k=1}^3 \frac{x^k}{k}, & \text{если } x \leq 0 \end{cases}$
8	Тема 8. Алгоритмы сортировки	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <p>1. Дана последовательность чисел a_1, a_2, \dots, a_n. Требуется расположить элементы по убыванию. Для этого в массиве, начиная с первого, выбирается наибольший элемент и ставится на первое место, а первый – на место наибольшего. Написать алгоритм сортировки выбором.</p> <p>2. Дана последовательность чисел a_1, a_2, \dots, a_n. Требуется представить числа в порядке возрастания. Для этого сравниваются два соседних числа a_i и a_{i+1}. Если $a_i > a_{i+1}$, то делается перестановка. Так продолжается до тех пор, пока все элементы не будут расположены в порядке возрастания. Составить алгоритм сортировки обменами, подсчитывая при этом количество перестановок.</p> <p>3. Дана последовательность чисел a_1, a_2, \dots, a_n. Требуется представить числа в порядке возрастания. Делается это следующим образом. Пусть a_1, a_2, \dots, a_i – упорядоченная последовательность, т.е. $a_1 \leq a_2 \leq \dots \leq a_i$. Берется следующее число a_{i+1} и вставляется в последовательность так, чтобы новая последовательность была тоже возрастающей. Процесс производится до тех пор, пока все элементы от $i+1$ до n не будут перебраны. Составить алгоритм сортировки обменами.</p> <p>4. Дан массив n действительных чисел. Требуется упорядочить его по возрастанию. Делается это следующим образом: сравниваются два соседних элемента a_i и a_{i+1}.</p>

		Если $a_i \leq a_{i+1}$, то продвигаются на один элемент вперед. Если $a_i > a_{i+1}$, то производится перестановка и сдвигаются на один элемент назад. Составить алгоритм сортировки Шелла.
9	Тема 9. Структуры данных	<p>Самостоятельное изучение тем, выполнение практических заданий.</p> <p>Примерная тематика практических заданий:</p> <p>1. Дан набор из 10 чисел. Создать две очереди: первая должна содержать числа из исходного набора с нечетными номерами (1, 3, ..., 9), а вторая — с четными (2, 4, ..., 10); порядок чисел в каждой очереди должен совпадать с порядком чисел в исходном наборе. Вывести указатели на начало и конец первой, а затем второй очереди.</p> <p>2. Создать стек целочисленных значений, для реализации используя односвязные списки. Реализовать операции добавления (push) и удаления (pop) элемента из стека. Добавьте в стек числа 4, 3, 1, 2, 4 и распечатайте содержимое стека. Удалите один элемент из стека, и распечатайте содержимое стека еще раз. Найдите минимальный элемент, принадлежащий стеку.</p> <p>3. Описать структуру с именем STUDENT, содержащую поля: Name – фамилия и инициалы, Kurs – курс, SES – успеваемость (массив из пяти элементов). Написать программу с использованием структуры, выполняющую:</p> <ul style="list-style-type: none"> – ввод с клавиатуры данных в массив STUD, состоящий из 10 структур типа STUDENT, записи должны быть упорядочены по алфавиту; – вывод на экран записей, упорядоченного списка студентов, средний бал которых превышает общий средний бал; – если таких студентов нет – выдать сообщение

5. Перечень основной и дополнительной учебной литературы

Основная литература:

1. Белов, В. В. Алгоритмы и структуры данных : учебник / В. В. Белов, В. И. Чистякова. - Москва : КУРС : ИНФРА-М, 2020. - 240 с. - (Бакалавриат). - ISBN 978-5-906818-25-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1057212> (дата обращения: 23.10.2020). – Режим доступа: по подписке.
2. Колдаев, В. Д. Основы алгоритмизации и программирования : учебное пособие / В. Д. Колдаев ; под ред. проф. Л. Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2021. — 414 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0733-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1151517> (дата обращения: 27.09.2021). – Режим доступа: по подписке.

Дополнительная литература:

1. Канцедал, С. А. Алгоритмизация и программирование : учебное пособие / С.А. Канцедал. — Москва : ФОРУМ : ИНФРА-М, 2021. —

- 352 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0727-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1189320> (дата обращения: 23.10.2020). – Режим доступа: по подписке.
2. Игнашева, Е. П. Системы счисления, алгоритмизация и программирование : учебное пособие / Е.П. Игнашева. — Москва : ИНФРА-М, 2020. — 224 с. — (Военное образование). - ISBN 978-5-16-015295-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1078360> (дата обращения: 23.10.2020). – Режим доступа: по подписке.
 3. Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. — Москва : ФОРУМ : ИНФРА-М, 2020. — 343 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-553-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1042452> (дата обращения: 23.10.2020). – Режим доступа: по подписке.
 4. Нидхем, М. Графовые алгоритмы : практическое руководство / М. Нидхем, Э. Ходлер ; пер. с англ. В. С. Яценкова. - Москва : ДМК Пресс, 2020. - 258 с. - ISBN 978-5-97060-799-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1094928> (дата обращения: 27.09.2021). – Режим доступа: по подписке.

Электронные книги:

1. Колдаев, В. Д. Основы алгоритмизации и программирования : учебное пособие / В. Д. Колдаев ; под ред. проф. Л. Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2021. — 414 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0733-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1151517> (дата обращения: 27.09.2021). – Режим доступа: по подписке.
2. Мишенин, А. И. Сборник задач по программированию : учебное пособие / А. И. Мишенин. - Москва : Финансы и Статистика, 2021. - 224 с. - ISBN 978-5-00184-039-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1541948> (дата обращения: 27.09.2021). – Режим доступа: по подписке.
3. Долгов, А. И. Алгоритмизация прикладных задач : учебное пособие / А. И. Долгов. - 3-е изд., стер. - Москва : Флинта, 2021. - 136 с. - ISBN 978-5-9765-0086-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1843168> (дата обращения: 27.09.2021). – Режим доступа: по подписке.

6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Интернет-ресурсы:

<http://www.intuit.ru/studies/courses/555/411/info> – Алгоритмы и теория вычислений.

<http://www.intuit.ru/studies/courses/1106/146/info> – Основы теории
вычислимых функций

<http://www.infojournal.ru> – Научно-образовательный портал,

<http://www.interface.ru/> – Научно-образовательный портал.

<http://www.codeblocks.org/>

<https://code.visualstudio.com/>

<http://www.cplusplus.com/> - Сеть ресурсов, посвященных языку C++

<http://pythontutor.ru/> - Курс по программированию на языке Python

<https://www.python.org/> – Сайт разработчиков Python

<https://jupyter.org/>

<https://colab.research.google.com/>

7. Перечень информационных технологий

Перечень программного обеспечения:

LibreOffice, VS Code, GCC + MinGW, Python 3, PyCharm

Информационные справочные системы:

1. Электронные ресурсы образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Язык программирования»