

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Смирнов Сергей Николаевич
Должность: врио ректора
Дата подписания: 26.10.2023 15:41:27
Уникальный программный ключ:
69e375c64f7e975d4e8830e7d11e224b159188

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Тверской государственный университет»

УТВЕРЖДАЮ

Руководитель ООП

С.М. Дудаков



Рабочая программа дисциплины (саннотацией)

Технология программирования

Направление подготовки

15.03.06 МЕХАТРОНИКА И РОБОТОТЕХНИКА

Профиль подготовки

Интеллектуальное управление в мехатронных и робототехнических системах

Для студентов 2-го курса

Формы обучения - очная

Составитель(и):

- к.т.н. доц. Волушкова В.Л.

I. Аннотация

1. Цель и задачи дисциплины:

дать общее представление обучающимся об объектно-ориентированном программировании, о методологии построения модели объектов в виде классов.

2. Место дисциплины в структуре ООП

Дисциплина входит в раздел «Информатика и информационно-коммуникационные технологии» обязательной части блока 1.

Предварительные знания и навыки. Знание языка программирования С и основных методов обработки информации: работа с текстом, сортировка, поиск.

Дальнейшее использование. Полученные знания используются в последующем при изучении предметов: «Базы данных», «Информационная безопасность», «Проектирование, разработка и эксплуатация информационных систем» и других.

3. Объем дисциплины: 7 зач. ед., 252 акад. часа, в том числе:

контактная аудиторная работа лекций 30 часов, практических занятий 30 часов, лабораторных занятий 30 часов,

контактная внеаудиторная работа контроль самостоятельной работы 10 часов, в том числе (расчетно-графическая) работа 10 часов;

самостоятельная работа 152 часа, в том числе контроль 36 часов.

4. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы:

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1, Анализирует задачу, выделяя ее базовые составляющие УК-1.2, Определяет, интерпретирует и ранжирует информацию, требуемую для решения поставленной задачи УК-1.3, Осуществляет поиск информации для решения поставленной задачи по различным типам запросов УК-1.4, При обработке информации отличает факты от мнений, интерпретаций, оценок, формирует собственные мнения и суждения, аргументирует свои выводы и точку зрения

	УК-1.5, Рассматривает и предлагает возможные варианты решения поставленной задачи, оценивая их достоинства и недостатки
ОПК-1, Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-1.1, Демонстрирует знания основ математики, физики, вычислительной техники и программирования
ОПК-11,	ОПК-11.2 Разрабатывает программное обеспечение для управления мехатронными и робототехническими системами
ОПК-14,	ОПК-14.2 Разрабатывает программное обеспечение для обработки информации в мехатронных и робототехнических системах
ПК-1,	ПК-1.2, Разрабатывает экспериментальные макеты управляющих, информационных и исполнительных модулей мехатронных и робототехнических систем и проводит их экспериментальное исследование с применением современных информационных технологий

5. Форма промежуточной аттестации и семестр прохождения:

РГР, экзамен в 3 семестре

6. Язык преподавания:

русский

II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Для студентов очной формы обучения

Учебная программа наименование разделов и тем	Всего (час.)	Контактная работа (час.)				Контроль сам. раб., в т.ч. кур- совая работа	Сам. раб., в т.ч. контроль (час.)
		Лекции		Практ. занятия / Лаб. работы			
		всего	в т.ч. практ подг.	всего	в т.ч. практ подг.		
1	2	3	4	5	6	7	8
Основные понятия и принципы объектно-ориентированного программирования (ООП).	39	4		4/4		0	27
Полиморфизм.	40	4		4/4		0	28
Реализация принципов объектно-ориентированного подхода в Java	40	6		6/6		0	22
Шаблоны проектирования	93	12		12/1 2		10	47
Потоки в Java.	40	4		4/4		0	28
Итого	252	30	0	30/3 0	0/0	10	152

Учебная программа дисциплины

- Основные понятия и принципы объектно-ориентированного программирования (ООП).
 - Понятие класса и объекта. Создание новых типов. Объявление класса. Определение объекта. Ограничение доступа к членам класса. Определение методов класса.
 - Конструкторы и деструкторы. Конструкторы и деструкторы, заданные по умолчанию.
 - Объявление функций-членов со спецификатором const. Чем отличается интерфейс от выполнения класса. Классы, содержащие другие классы в качестве данных-членов.
 - Роль инкапсуляции в ООП. Два взгляда на инкапсуляцию.
 - Инкапсуляция и скрытие реализации. Скрытие методов и допустимые состояния объектов.
 - Механизмы обхода инкапсуляции.
- Полиморфизм.
 - Роль полиморфизма в ООП. Виды полиморфизма.
 - Специальный полиморфизм. Перегрузка функций. Дополнительные сведения о функциях.
 - Чистый полиморфизм, его роль и реализация. Механизм вызова виртуальных методов.

- Шаблонный полиморфизм. Примеры применения чистого и шаблонного полиморфизма.

3. Реализация принципов объектно-ориентированного подхода в Java

- Типы данных, переменные, массивы. Классы.
- Класс Object. Пакеты.
- Модификаторы доступа. Область видимости классов, методов, переменных.
- Наследование. Классы, суперклассы и подклассы.
- Абстрактные классы и интерфейсы.
- Динамическое связывание. Терминальные классы и методы.
- Обработка событий в Java
- Интерфейсы и обратный вызов
- Безымянный внутренние классы. Определение изменения состояния объекта.
- Модель делегирования событий в Java.
- Иерархия событий библиотеки AWT
- Графические интерфейсы пользователя в Java.
- Обзор пакета Java.awt.
- Система Swing.
- Построение меню.
- Создание диалоговых окон.

4. Шаблоны проектирования

- Порождающие паттерны, UML- диаграмма классов и диаграмма последовательности.
- Структурные паттерны: декоратор, фасад
- Поведенческие паттерны: команда, стратегия, наблюдатель, шаблонный метод.
- Архитектура «модель-представление-контроллер».

5. Потоки в Java.

- Периоды жизни потока. Класс Thread. Интерфейс Runnable.
- Способы организации потоков в java-кодах.

III. Образовательные технологии

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Основные понятия и принципы объектно-ориентированного программирования (ООП).	лекции, практические занятия, лабораторные занятия	изложение теоретического материала, решение задач, работа с программным обеспечением, разработка программного обеспечения
Полиморфизм.	лекции, практические занятия, лабораторные занятия	изложение теоретического материала, решение задач, работа с программным обеспечением, разработка программного обеспечения
Реализация принципов объектно-ориентированного подхода в Java	лекции, практические занятия, лабораторные занятия	изложение теоретического материала, решение задач, работа с программным обеспечением, разработка программного обеспечения
Шаблоны проектирования	лекции, практические занятия, лабораторные занятия	изложение теоретического материала, решение задач, выполнение РГР, работа с программным обеспечением, разработка программного обеспечения
Потоки в Java.	лекции, практические занятия, лабораторные занятия	изложение теоретического материала, решение задач, работа с программным обеспечением, разработка программного обеспечения

IV. Оценочные материалы для проведения текущей и промежуточной аттестации

Типовые контрольные задания и/или критерии для проверки индикаторов ОПК-1.1, ОПК-11.1, ОПК-14.1

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
---------------------------	--	--

<p>Знает современные технологии и программные средства. Владеет навыками применения современных технологий</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Создать класс окружность, содержащий поля, в которых можно хранить координаты и радиус. Класс должен содержать три конструктора – по умолчанию, конструктор с параметрами, конструктор копирования. Для каждого поля написать методы set и get. Перегрузить операторы >, <(по площади) и ==, оператор потокового вывода <. Предусмотреть все возможные проверки. Написать функцию main, демонстрирующую возможности класса. • Создать класс трамвайный билет, содержащий поля, в которых можно хранить шестизначный номер и серию. Класс должен содержать два конструктора – по умолчанию, конструктор с параметрами. Для каждого поля написать методы set и get. Перегрузить операторы >, <(по номеру) и ==, 	<p>оценка 3 — Знает как создавать классы и объекты. Умеет создать класс, для описания конкретного понятия. оценка 4 — кроме того умеет создать конструктор и деструктор класса, инкапсулировать данные класса, оценка 5 — кроме того умеет создать конструктор копирования и продемонстрировать его работу.</p>
<p>Требования к обучающемуся</p>	<p>Типовые контрольные задания для оценки знаний, умений, навыков</p>	<p>Показатели и критерии оценивания, шкала оценивания</p>
	<p>метод «счастливый билет» (сумма первых трех цифр равна сумме последних), оператор потокового вывода <<. Предусмотреть все возможные проверки. Написать функцию main, демонстрирующую возможности класса.</p> <ul style="list-style-type: none"> • Создать класс треугольник, содержащий поля, в которых можно хранить координаты вершин. Класс должен содержать два конструктора – по умолчанию, конструктор с параметрами. Перегрузить операторы >, <(по площади) и ==, оператор потокового вывода <<. Предусмотреть все возможные проверки. Написать функцию main, демонстрирующую возможности класса. 	

<p>Знает современные технологии программные средства. Владеет навыками применения современных технологий. Умеет самостоятельно создавать методы класса, используя понятие полиморфизма и знания о виртуальных функциях.</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Создайте базовый класс список и производные классы стек и очередь. Определить функцию вставки элемента в стек и очередь. Используйте Java. • Создайте базовый класс средства транспорта и производные классы - автомобиль, лодка, велосипед. Определите функции движения и подсчёта стоимости всех транспортных средств семьи. Используйте Java. • Создать иерархию типов - бытовая техника, кухонная техника, телевизионная техника, кухонный комбайн с монитором. Классы должны содержать конструкторы, включая, конструктор копирования, виртуальные деструкторы, перегруженные функции вывода в поток и ввода в поток. Используйте C++ и множественное наследование. 	<p>оценка 3 — умеет использовать понятие наследования для описания иерархии классов, умеет заместить методы базового класса. оценка 4 — кроме того умеет использовать виртуальные методы и реализовывать их в C++ и Java, оценка 5 — кроме того умеет использовать статические функции и данные, абстрактные классы и множественное наследование в C++.</p>
<p>Может выбирать современные технологии для описания коллекций. Умеет самостоятельно создавать классы для описания коллекций, используя шаблонный полиморфизм.</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Создать шаблонный класс массив. Класс должен содержать три конструктора – по умолчанию, конструктор с параметрами, конструктор копирования. Перегрузить операции ==, =, [], +, оператор потокового вывода «. Использовать C++. • Создать шаблонный класс строка. Класс должен содержать три конструктора – по умолчанию, конструктор с параметрами, конструктор копирования. Перегрузить операции ==, =, [], +, >, <, оператор потокового вывода «. Использовать C++. • Создать класс список. Использовать дженерики. Класс должен содержать два конструктора – по умолчанию, конструктор с 	<p>оценка 3 — Может выбрать подход, основанный на шаблонном полиморфизме, для создания коллекций, оценка 4 — кроме того, умеет создать класс, описывающий коллекцию, оценка 5 — кроме того умеет</p>

	<p>параметрами. Предусмотреть методы клонирования, сравнения, добавления эл-та в список, получения эл-та из списка по индексу. Перегрузить метод toString(). Использовать java.</p>	<p>перегрузить операторы ==, =, [] для C++ и методы toString() и equal() для Java.</p>
<p>Знает современные технологии и программные средства. Знает понятие шаблона проектирования, порождающие паттерны, UML-диаграмма классов и диаграмма последовательности.</p>	<p>Примеры вопросов к экзамену:</p> <ul style="list-style-type: none"> • Виды шаблонов проектирования. Порождающие паттерны. • UML- диаграмма классов и диаграмма последовательности. Привести примеры этих диаграмм для паттерна фабричный метод. • Порождающий паттерн одиночка. UML- диаграмма. • Паттерн прототип. UML- диаграмма. 	<p>оценка 3 — знает понятие паттерн проектирования и порождающие паттерны. оценка 4 — знает что такое UML- диаграмма классов и диаграмма последовательности, оценка 5 — знает что такое паттерн фабричный.</p>
<p>Знает современные технологии и программные средства. Знает понятие структурные паттерны: декоратор, фасад</p>	<p>Примеры вопросов к экзамену:</p> <ul style="list-style-type: none"> • Паттерн декоратор. UML- диаграмма. Примеры. • Паттерн фасад. UML- диаграмма. Примеры. 	<p>оценка 3 — знает понятие структурные паттерны оценка 4 — кроме того может составить UML- диаграмму структурного паттерна, оценка 5 — кроме того может продемонстрировать на примере работу структурного паттерна.</p>

<p>Знает современные технологии и программные средства. Знает понятие поведенческие паттерны: команда, стратегия, наблюдатель, шаблонный метод</p>	<p>вопросов к экзамену:</p> <ul style="list-style-type: none"> • Паттерн команда. UML- диаграмма. Примеры. • Паттерн стратегия. UML- диаграмма. Примеры. • Паттерн наблюдатель. UML- диаграмма. Примеры. • Паттерн шаблонный метод. UML- диаграмма. Примеры. 	<p>оценка 3 — знает понятия поведенческий паттерн оценка 4 — кроме того может создать UML-диаграмму поведенческого паттерна, оценка 5 — кроме того может продемонстрировать на примере работу поведенческого паттерна.</p>
--	--	--

Типовые контрольные задания и/или критерии для проверки индикаторов УК-1.1, УК-1.3, УК-1.4, УК-1.5

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
<p>Способен анализировать задачу, разрабатывать алгоритмы и программы, пригодные для практического применения. Умение анализировать предметную область и описывать ее с помощью классов, поддерживающих</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Создайте иерархию классов, описывающую книгу. У книги есть автор, название, количество страниц и цена. Ваша программа должна выводить информацию о книге, причём цена должна быть в долларах или в евро. При изображении цены в долларах перед числом должен стоять значок (\$) и описание (34 доллара). При изображении цены в евро после числа должен быть текст «eur» (34eur) и описание (34 евро). • Создать иерархию классов, описывающих создателей книги. В качестве создателей выступает писатель и художник. Ваша программа должна в одной и той же переменной сохранять одного из двух создателей книги. Продемонстрируйте метод создания книги различными создателями. (писатель – «write pen the book» художник «Brush paint the book»). 	<p>оценка 3 — умеет создать алгоритм, основанный на иерархии классов, для решения конкретной задачи. оценка 4 — кроме того умеет правильно использовать наследование или композицию. оценка 5 — кроме того умеет использовать паттерн "стратегия".</p>

динамическое связывание,		
Способен осуществлять поиск информации для решения поставленной задачи, разрабатывать алгоритмы и программы, пригодные для практического применения. Умение обрабатывать события в Java, создавать интерфейсы, безымянные внутренние классы и использовать модель делегирования событий.	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Создайте программу и протестируйте «рисования» примитивных фигур в Java. Используйте интерфейс, который должен содержать методы show() и hide(). • Создайте программу обработки события Timer в Java. Используйте безымянные внутренние классы. Через определённое время timer должен выводить модальное окно с сообщением. • Создайте программу обработки события «нажать на кнопку». Используйте безымянные внутренние классы. При нажатии на кнопку должно выводиться модальное окно с сообщением. 	<p>оценка 3 — умеет создавать, отлаживать и тестировать программы, обрабатывающие события в Java.</p> <p>оценка 4 — кроме того имеет представление о безымянных внутренних их классах,</p> <p>оценка 5 — кроме того умеет обрабатывать события «мышь».</p>

Типовые контрольные задания и/или критерии для проверки индикаторов ПК-1.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
---------------------------	--	--

<p>Знает основные технологии создания современных программных систем.</p> <p>Может разрабатывать экспериментальные макеты управляющих, информационных и исполнительных модулей мехатронных и робототехнических систем.</p>	<p>Примеры тем для расчётно-графической работы:</p> <p>Примеры задач для самостоятельных работ (задание для групповой работы):</p> <p>Создать и протестировать элементарный графический редактор, рисующий стандартные фигуры-прямоугольник, овал, прямоугольник. Предусмотреть возможность добавления фигуры, стирания фигуры, перемещения фигуры, сохранения фигуры в файле и восстановления фигуры, хранящиеся в файле. Реализовать архитектуру «модель-представление-контроллер».</p> <p>Использовать пакет Java swing.</p> <ul style="list-style-type: none"> • Создать иерархию классов для описания редактора. (выполняет руководитель группы) • Разбить на этапы работу по созданию редактора (выполняет руководитель группы) • Распределить среди участников группы задания, которые необходимо выполнить для решения поставленной задачи (выполняет руководитель группы) • Создать классы для описания фигуры (выполняет участник группы) • Создать классы для поддержки меню и обработки событий (выполняет участник группы) • Создать классы для поддержки сохранения информации и (выполняет участник группы) • Создать тесты, которые проверяют правильность работы системы (выполняет участник группы) • Выполнить тестирование редактора (выполняют все участники группы) 	<p>оценка 3 — руководитель группы создал иерархию классов для описания редактора и смог разбить работу группы на этапы, участник группы создал основные классы или тесты. оценка 4 — разработанное приложение выполняет основные функции. оценка 5 кроме выполнено полное тестирование</p>
--	---	--

Важной составляющей данного раздела РПД являются требования к рейтинг-контролю с указанием баллов, распределенных между модулями и видами работы обучающихся.

Максимальная сумма баллов по учебной дисциплине, заканчивающейся экзаменом, по итогам семестра составляет 60 баллов (30 баллов - 1-й модуль и 30 баллов - 2-й модуль).

Обучающемуся, набравшему 40–54 балла, при подведении итогов семестра (на последнем занятии по дисциплине) в рейтинговой ведомости учета успеваемости и зачетной книжке может быть выставлена оценка «удовлетворительно».

Обучающемуся, набравшему 55–57 баллов, при подведении итогов семестра (на последнем занятии по дисциплине) в графе рейтинговой ведомости учета успеваемости «Премияльные баллы» может быть добавлено 15 баллов и выставлена экзаменационная оценка «хорошо».

Обучающемуся, набравшему 58–60 баллов, при подведении итогов семестра (на последнем занятии по дисциплине) в графе рейтинговой ведомости учета успеваемости «Премияльные баллы» может быть добавлено 27 баллов и выставлена экзаменационная оценка «отлично». В каких-либо иных случаях добавление премиальных баллов не допускается.

Обучающийся, набравший до 39 баллов включительно, сдает экзамен.

Распределение баллов по модулям устанавливается преподавателем и может корректироваться.

V. Учебно-методическое и информационное обеспечение дисциплины

1. Рекомендованная литература

а) Основная литература

[1] Дэвид, Х. Разработка приложений Java EE 6 в NetBeans 7. [Электронный ресурс] : Самоучители и руководства — Электрон. дан. — М. : ДМК Пресс, 2013. — 330 с. — Режим доступа: <http://e.lanbook.com/book/58693> — Загл. с экрана.

[2] Гаврилов, А.В. Программирование на Java. Конспект лекций. [Электронный ресурс] : Справочники / А.В. Гаврилов, С.В. Клименков, Е.А. Цопа. — Электрон. дан. — СПб. : НИУ ИТМО, 2010. — 130 с. — Режим доступа: <http://e.lanbook.com/book/43547> — Загл. с экрана.

б) Дополнительная литература

[3] Современные Java-технологии на практике: Практическое руководство / Машнин Т.С. - СПб:БХВ-Петербург, 2010. - 560 с. ISBN 978-5-9775-0561-1 — Режим доступа: <http://znanium.com/catalog.php?bookinfo=351236> — Загл. с экрана.

2. Программное обеспечение

Компьютерный класс факультета прикладной математики и кибернетики № 46 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	
Adobe Acrobat Reader DC - Russian	бесплатно
Apache Tomcat 8.0.27	бесплатно
Cadence SPB/OrCAD 16.6	Государственный контракт на поставку лицензионных программных продуктов 103 - ГК/09 от 15.06.2009
GlassFish Server Open Source Edition 4.1.1	бесплатно
Google Chrome	бесплатно
Java SE Development Kit 8 Update 45	бесплатно

(64-bit)	
JetBrains PyCharm Community Edition 4.5.3	бесплатно
JetBrains PyCharm Edu 3.0	бесплатно
Kaspersky Endpoint Security 10 для Windows	Акт на передачу прав ПК545 от 16.12.2022
Lazarus 1.4.0	бесплатно
Mathcad 15 Мо10	Акт предоставления прав ИС00000027 от 16.09.2011
MATLAB R2012b	Акт предоставления прав № Ус000311 от 25.09.2012
Многофункциональный редактор ONLYOFFICE бесплатное ПО	бесплатно
ОС Linux Ubuntu бесплатное ПО	бесплатно
MiKTeX 2.9	бесплатно
MSXML 4.0 SP2 Parser and SDK	бесплатно
NetBeans IDE 8.0.2	бесплатно
NetBeans IDE 8.2	бесплатно
Notepad++	бесплатно
Oracle VM VirtualBox 5.0.2	бесплатно
Origin 8.1 Sr2	договор №13918/М41 от 24.09.2009 с ЗАО «СофтЛайн Трейд»
Python 3.1 pygame-1.9.1	бесплатно
Python 3.4 numpy-1.9.2	бесплатно
Python 3.4.3	бесплатно
Python 3.5.1 (Anaconda3 2.5.0 64-bit)	бесплатно
WCF RIA Services V1.0 SP2	бесплатно
WinDjView 2.1	бесплатно
R Studio	бесплатно
Anaconda3 2019.07 (Python 3.7.3 64-bit)	бесплатно

Ауд. 201а (компьютерная лаборатория ПМиК) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Перечень программного обеспечения (со свободными лицензиями): Linux Kubuntu, KDE, TeXLive, TeXStudio, LibreOffice, GIMP, Gwenview, ImageMagick, Okular, Skanlite, Google Chrome, KDE Connect, Konversation, KRDC, KTorrent, Thunderbird, Elisa, VLC media player, PulseAudio, KAppTemplate, KDevelop, pgAdmin4, PostgreSQL, Qt, QtCreator, R, RStudio, Visual Studio Code, Perl, Python, Ruby, clang, clang++, gcc, g++, nasm, flex, bison, Maxima, Octave, Dolphin, HTop, Konsole, KSystemLog, Xterm, Ark, Kate, KCalc, Krusader, Spectacle, Vim
--	---

3) Современные профессиональные базы данных и информационные справочные системы

1. ЭБС «ZNANIUM.COM» www.znanium.com;
2. ЭБС «Университетская библиотека онлайн» <https://biblioclub.ru/>;
3. ЭБС «Лань» <http://e.lanbook.com>.

4) Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины
Интернет-университет <http://www.intuit.ru>

VI. Методические материалы для обучающихся по освоению дисциплины

Примеры задач для подготовки к контрольным работам

1. Создайте класс стек в Java. Проанализируйте с его помощью баланс скобок «(», «)» во введенной строке.
2. Создайте класс массив в C++, в котором предусмотрите методы, которые будут выполнять следующее: сортировать массив по заданным значениям полей. получать элемент массива, задавать элемент массива, добавлять элемент в массив, распечатывать массив, заполнять массив данными из текстового файла.
3. Создайте программу «рисования» примитивных фигур в Java. Используйте интерфейс, который должен содержать методы show() и hide().
4. Создайте базовый класс список и производные классы стек и очередь. Определите функцию вставки элемента в стек и очередь. Вставьте элемент во все стеки и очереди, имеющиеся в вашей программе. Используйте C++.
5. Создайте базовый класс средства транспорта и производные классы - автомобиль, лодка, велосипед. Определите функции движения и подсчета стоимости всех транспортных средств семьи. Используйте C++.
6. Создать иерархию типов - бытовая техника, кухонная техника, телевизионная техника, кухонный комбайн с монитором. Классы должны содержать конструкторы, включая, конструктор копирования, виртуальные деструкторы, перегруженные функции вывода в поток и ввода в поток. Используйте C++ и множественное наследование.

Требования к рейтинг контролю

Контрольная работа 1. Темы: виртуальные методы java. Пример задания:

1. Создать базовый класс список и производные классы стек и очередь. Определить функцию вставки и получения элемента для стека и очереди. В вызывающей программе создайте массив из стека и очереди. Вставьте элемент во все стеки и очереди, имеющиеся в вашей программе.
2. Напишите класс “параметризованный стек” в Java. Разместите класс стек и основной класс в разных файлах. Реализуйте методы всех арифметических операций, а также модифицируйте функцию toString() и public boolean equals(Object O). Обработайте возможные ошибки. Упакуйте созданный класс «стек» в архив jar.

За решение каждой задачи выставляется максимум 10 баллов.

Контрольная работа 2. Темы: Шаблоны проектирования. Пример задания:

1. Создайте иерархию классов, в которой метод execute(int a,int b) одного из классов

выполнял одно из арифметических действий(+,-,*,/). Определение действия задается в конструкторе класса.

2. Напишите класс, функция которого может читать текстовые данные вещественное число из файла и с экрана и выводить среднее арифметическое этих чисел. (Шаблонный метод)

За решение каждой задачи выставляется максимум 10 баллов.

Самостоятельная работа. Темы: Использование шаблонов проектирования. Пример задания:

1. Создайте элементарный графический редактор, рисующий стандартные фигуры- прямую, овал, прямоугольник. Предусмотрите возможность добавлять фигуры, стирать фигуры, перемещать фигуры, сохранять фигуры в файле и восстанавливать фигуры, хранящиеся в файле. Использовать рассмотренные паттерны проектирования. Использовать пакет Java swing.
2. Задача обедающих философов сформулированная Э. Дейкстрой в 1965 г. Пять философов сидят за круглым столом. У каждого философа есть тарелка со спагетти. Каждому философу для еды нужны две вилки. Между каждыми двумя тарелками лежит одна вилка. Жизненный цикл каждого философа - последовательность двух чередующихся состояний: прием пищи и размышление. Голодный философ пытается получить две вилки, левую и правую, в любом порядке. В каждый момент времени философ может захватить только одну вилку. Философ, захвативший две вилки, некоторое время ест, затем освобождает вилки и продолжает размышление. Предложить алгоритм, который моделирует эти действия для каждого философа и никогда не застревает. Отобразить состояния философов в графическом виде.

За выполнение самостоятельной работы выставляется максимум 20 баллов.

Общая сумма В сумме за все задачи выставляет не более 40 баллов.

За выполнение самостоятельной работы выставляется максимум 20 баллов. За ответ на экзамене выставляется максимум 40 баллов.

VII. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Для аудиторной работы

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 206 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, экран, проектор.

Ауд. 4б (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, экран, компьютер, проектор, МФУ.
--	---

Для самостоятельной работы

Наименование помещений	Материально-техническое оснащение помещений
Компьютерная лаборатория факультета ПМиК № 201а	Набор учебной мебели, доска маркерная, компьютер, сервер (системный блок), концентратор сетевой.

VIII. Сведения об обновлении рабочей программы дисциплины

№ п.п.	Обновленный раздел рабочей программы дисциплины	Описание внесенных изменений	Дата и протокол заседания кафедры, утвердившего изменения
1	V. Учебно-методическое и информационное обеспечение дисциплины 2) Программное обеспечение	Внесены изменения в программное обеспечение	От 24.08.2023 года, протокол № 1 ученого совета факультета