ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Иркутский государственный университет путей сообщения»
(ФГБОУ ВО ИрГУПС)

УТВЕРЖДЕНА
приказом и.о. ректора
от «08» мая 2020 г. № 267-1

Б1.О.26 Объектно-ориентированное программирование

рабочая программа дисциплины

Специальность/направление подготовки – 09.03.04 Программная инженерия
Специализация/профиль – Разработка программно-информационных систем
Квалификация выпускника – Бакалавр
Форма и срок обучения – очная форма 4 года
Кафедра-разработчик программы – Информационные системы и защита информации

Общая трудоемкость в з.е. – 9
Часов по учебному плану (УП) – 324

Формы промежуточной аттестации
очная форма обучения:
зачет 2 семестр, экзамен 3 семестр

Очная форма обучения **Распределение часов дисциплины по семестрам**

Семестр	2	3	**Итого**
Вид занятий	Часов по УП	Часов по УП	**Часов по УП**
Аудиторная контактная работа по видам учебных занятий/ в т.ч. в форме ПП*	68	85	**153**
– лекции	34	34	**68**
– практические (семинарские)		17	**17**
– лабораторные	34	34	**68**
Самостоятельная работа	76	59	**135**
Экзамен		36	**36**
Итого	144	180	**324**

ИРКУТСК

Электронный документ выгружен из ЕИС ФГБОУ ВО ИрГУПС и соответствует оригиналу
Подписант ФГБОУ ВО ИрГУПС Трофимов Ю.А.
00920FD815CE68F8C4CA795540563D259C с 07.02.2024 05:46 по 02.05.2025 05:46 GMT+03:00
Подпись соответствует файлу документа

Рабочая программа дисциплины разработана в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по направлению подготовки 09.03.04 Программная инженерия, утвержденным Приказом Министерства науки и высшего образования Российской Федерации от 19.09.2017 № 920.

Программу составил(и):
д.т.н., профессор, профессор, Е.И. Молчанова

Рабочая программа рассмотрена и одобрена для использования в учебном процессе на заседании кафедры «Информационные системы и защита информации», протокол от «29» апреля 2020 г. № 11

Зав. кафедрой, д.т.н., доцент Л.В. Аршинский

1 ЦЕЛИ И ЗАДАЧА ДИСЦИПЛИНЫ	
1.1 Цели дисциплины	
1	овладение теоретическими и прикладными знаниями и умениями в области объектно-ориентированного анализа предметной области
2	освоение технологии объектно-ориентированного программирования для решения практических задач информатики, необходимых для осуществления профессиональной деятельности в проектной и производственно-технологической сфере
1.2 Задача дисциплины	
1	приобретение навыков самостоятельного и творческого использования теоретических знаний в практической деятельности по производству программно-информационных систем различного назначения
1.3 Цель воспитания и задачи воспитательной работы в рамках дисциплины	
Профессионально-трудовое воспитание обучающихся	
Цель – формирование у обучающихся осознанной профессиональной ориентации, понимания общественного смысла труда и значимости его для себя лично, ответственного, сознательного и творческого отношения к будущей деятельности, профессиональной этики, способности предвидеть изменения, которые могут возникнуть в профессиональной деятельности, и умению работать в изменённых, вновь созданных условиях труда. Цель профессионально-трудового воспитания достигается по мере решения в единстве следующих задач: – формирование сознательного отношения к выбранной профессии; – воспитание чести, гордости, любви к профессии, сознательного отношения к профессиональному долгу, понимаемому как личная ответственность и обязанность; – формирование психологии профессионала; – формирование профессиональной культуры, этики профессионального общения; – формирование социальной компетентности и другие задачи, связанные с имиджем профессии и авторитетом транспортной отрасли	

2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП	
Блок/часть ОПОП	Блок 1. Дисциплины / Обязательная часть
2.1 Дисциплины и практики, на которых основывается изучение данной дисциплины	
1	Б1.О.33 Схемотехнические основы программно-вычислительных систем
2	Б2.О.01(У) Учебная - ознакомительная практика
2.2 Дисциплины и практики, для которых изучение данной дисциплины необходимо как предшествующее	
1	Б1.О.13 Математическая логика и теория алгоритмов
2	Б1.О.18 Программирование
3	Б1.О.21 Операционные системы
4	Б1.О.22 Базы данных
5	Б1.О.24 Компьютерные сети
6	Б1.О.25 Тестирование и отладка программного обеспечения
7	Б1.О.27 Надежность программного обеспечения
8	Б1.О.29 Вычислительные алгоритмы
9	Б1.О.30 Администрирование программно-информационных систем
10	Б1.О.31 Теория языков программирования и методы трансляции
11	Б1.О.32 Машинно-зависимые языки программирования
12	Б1.О.36 Проектирование программного обеспечения
13	Б2.О.01(У) Учебная - ознакомительная практика
14	Б2.О.02(У) Учебная - эксплуатационная практика
15	Б2.О.03(П) Производственная - технологическая (проектно-технологическая) практика
16	Б2.О.04(Пд) Производственная - преддипломная практика
17	Б3.01(Д) Выполнение выпускной квалификационной работы
18	Б3.02(Д) Защита выпускной квалификационной работы

3 ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ТРЕБОВАНИЯМИ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения
ОПК-2 Способен понимать принципы работы современных информационных	ОПК-2.1 Знает современные информационные технологии и программные средства, в том числе	Знать: современные технологии разработки ПО (структурное, объектно- ориентированное)
		Уметь: использовать современные технологии разработки ПО

технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	отечественного производства при решении задач профессиональной деятельности	Владеть: навыками использования современных технологий разработки ПО
	ОПК-2.2 Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	Знать: характеристики современных информационных технологий и программных средств, в том числе отечественного производства
		Уметь: формулировать требования к программным средствам, в том числе отечественного производства при решении задач профессиональной деятельности
		Владеть: навыками сопоставления характеристик современных информационных технологий и программных средств
	ОПК-2.3 Имеет навыки применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности	Знать: современные объектно-ориентированные среды программирования
		Уметь: разрабатывать объектно-ориентированное программное обеспечение
		Владеть: навыками объектного моделирования предметной области
ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	ОПК-5.1 Знает основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем	Знать: современные стандарты информационного взаимодействия систем
		Уметь: создавать программные интерфейсы
		Владеть: навыками в создании современных программных интерфейсов
	ОПК-5.2 Умеет выполнять параметрическую настройку информационных и автоматизированных систем	Знать: способы параметрической настройки информационных систем
		Уметь: выполнять параметрическую настройку информационных систем
		Владеть: навыками настройки программного обеспечения
	ОПК-5.3 Имеет навыки инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем	Знать: методики инсталляции программного обеспечения
		Уметь: инсталлировать программы и программные системы
		Владеть: навыками инсталляции программного обеспечения
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий	Знать: объектно-ориентированный язык программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий
		Уметь: современные программные среды разработки информационных систем и технологий
		Владеть: навыками применения основ информатики и программирования к проектированию, конструированию и тестированию программных продуктов
	ОПК-6.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	Знать: современные языки объектно-ориентированного программирования
		Уметь: применять современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач
		Владеть: навыками решения прикладных задач различных классов

	ОПК-6.3 Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач	Знать: объектно-ориентированные языки программирования и технологии отладки и тестирования прототипов программно-технических комплексов задач
		Уметь: разрабатывать программно-технические комплексы задач в объектно-ориентированной парадигме
		Владеть: навыками объектно-ориентированного программирования, отладки и тестирования прототипов программно-технических комплексов

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Очная форма					*Код индикатора достижения компетенции
		Семестр	Часы				
			Лек	Пр	Лаб	СР	
1.0	**Раздел 1. Основы языка программирования Java. Программы. Данные. Организация данных.**						
1.1	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 1	2				50	ОПК-2.1 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
1.2	Тема 1: Введение. Технологии Java SE, JEE. История и место Java среди современных языков программирования. Основные термины и инструментарий разработчика. Запуск приложения в командной строке. Передача параметров через командную строку	2	1				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
1.3	Тема 2: Знакомство со средой программирования NetBeans IDE. Установка среды. Структура кода простейшего приложения Java. Определение и вызов метода. Компиляция файлов проекта и запуск приложения. Структура проекта NetBeans. Отладка	2	1		5		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
1.4	Тема 3: Общие представления о языке Java. Алфавит Константы. Специальные символы. Управляющие конструкции. Составной оператор. Условный оператор if . Оператор выбора switch. Условное выражение …?... : Особенность операторов инкремента ++ и декремента Оператор цикла с пареметром for. Оператор цикла с предусловием while. Оператор цикла с постусловием do...while.	2	4		6		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
1.5	Тема 4: Оперативное хранение данных в языке Java. Примитивные типы переменных. Приведение числовых типов. Ссылочные переменные	2	3				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
1.6	Тема 5: Ссылочные переменные и объекты. Процедурное и объектно-ориентированное программирование. Классы,	2	4				ОПК-2.1 ОПК-2.2 ОПК-2.3

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ							
Код	Наименование разделов, тем и видов работ	Очная форма					*Код индикатора достижения компетенции
		Семестр	Часы				
			Лек	Пр	Лаб	СР	
	объекты и объектные переменные. Классы для типов данных и структур данных						ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
1.7	Тема 6: Массивы как объекты. Описание. Длина массива. Многомерные массивы. Зубчатый массив. Сравнение и копирование массивов. Класс Arrays. Оператор цикла for для перебора элементов массива (foreach)	2	4		4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.2 ОПК-6.3
1.8	Тема 7: Работа со строками Классы для работы со строками String, StringBuffer, StringBuilder. Создание строк и конкатенация. Методы для манипулирования строками. Консольный ввод. Форматированный вывод	2	4		4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
2.0	**Раздел 2. Объектная модель языка Java.**						
2.1	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 2	2				30	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
2.2	Тема 8: Введение в объектно-ориентированное программирование. Объектно-ориентированная парадигма программирования. Основные понятия объектного подхода	2	1				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-6.1 ОПК-6.2 ОПК-6.3
2.3	Тема 9: Объектная модель языка Java. Объявление классов. Заголовок класса. Модификаторы. Пакеты. Ключевое слово class. Ключевое слово extends. Ключевое слово implements. Тело класса. Объявление полей. Объявление методов. Объявление конструкторов. Объектные инициализаторы. Перегрузка (overload). Методы с переменным числом параметров	2	6		7		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
2.4	Тема 10: Базовые принципы ОО подхода. Абстракция и инкапсуляция. Наследование и полиморфизм. Использование super и this. Переопределение методов. Абстрактные классы. Интерфейсы. Клонирование объектов	2	6		6		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
2.5	Тема 11: Ошибки при работе программы. Исключения (Exceptions). Конструкция try-catch-finally. Использование	2	2		2		ОПК-2.1 ОПК-2.2

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ							
Код	**Наименование разделов, тем и видов работ**	**Очная форма**					***Код индикатора достижения компетенции**
		Семестр	**Часы**				
			Лек	**Пр**	**Лаб**	**СР**	
	операторов throw, throws. Классы исключений. Создание пользовательских классов исключений						ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
	Форма промежуточной аттестации – зачет	2					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
3.0	**Раздел 3. Введение в проектирование классов.**						
3.1	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 3	3				13	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
3.2	Тема 12: Объектно-ориентированный анализ. Язык UML.Рекомендации по проектированию классов. Класс Object	3	2	5	4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
3.3	Тема 13: Вложенные типы. Внутренние, вложенные, анонимные классы	3	4		4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
4.0	**Раздел 4. Использование классов и библиотек.**						
4.1	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 4	3				42	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
4.2	Тема 14. Потоки выполнения. Класс Thread и интерфейс Runnable. Потоки-демоны	3	4		4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ							
Код	**Наименование разделов, тем и видов работ**	**Очная форма**					***Код индикатора достижения компетенции**
		Семестр	**Часы**				
			Лек	**Пр**	**Лаб**	**СР**	
							ОПК-6.1 ОПК-6.2 ОПК-6.3
4.3	Тема 15. Потоки ввода/вывода. Байтовые и символьные потоки. Класс File. Вывод/Ввод текстовой информации. Класс RandomAccessFile. Сериализация объектов Класс Scanner	3	4	4	4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
4.4	Тема 16. Параметризация типов и ее применение. Обобщения (Generic). Параметризированные типы. Метасимвол «?»..Ограничения. Параметризованные методы	3	4		4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
4.5	Тема 17. Каркас коллекций. Очереди, стеки, списки, множества. Карты отображений	3	4		4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
4.6	Тема 18. Введение в программирование GUI средствами Swing. Связь с архитектурой MVC. Размещение компонентов. Обработка событий. Рисование средствами Swing	3	6	4	6		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
4.7	Тема 19. JDBC. Драйверы, соединения и запросы	3	4	4	4		ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
	Форма промежуточной аттестации – экзамен	3	36				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3
	Итого часов (без учёта часов на промежуточную аттестацию)		68	17	68	135	

5 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ
Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине оформлен в виде приложения № 1 к рабочей программе дисциплины и размещен в электронной информационно-образовательной среде Университета, доступной обучающемуся через его личный кабинет

6 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ		
6.1 Учебная литература		
6.1.1 Основная литература		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.1.1	Мейер, Б. Объектно-ориентированное программирование и программная инженерия : учебное пособие - 2-е изд., испр. / Б. Мейер. Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 286с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=429034 (дата обращения: 10.05.2024))	Онлайн
6.1.2 Дополнительная литература		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.2.1	Объектно-ориентированное программирование : учебное пособие : в 3 частях : [16+] / П. П. Степанов, А. А. Кабанов, В. А. Никонов, Т. С. Павлюченко ; ред. К. В. Обухова ; Омский государственный технический университет. – Омск : Омский государственный технический университет (ОмГТУ), 2021. – Часть 1. – 112 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=700657 (дата обращения: 10.05.2024).	Онлайн
6.1.2.2	Молчанова, Елена Ивановна Объектно-ориентированное программирование. Основы объектного программирования на языке Java в среде IDE NetBeans учеб. пособие : в 2 ч. : учеб. пособие : в 2 ч. / Е. И. Молчанова, В. В. Федоров ; Федер. агентство ж.-д. трансп., Иркут. гос. ун-т путей сообщ.. Иркутск : ИрГУПС, 2022. - 128с.	21
6.1.3 Учебно-методические разработки (в т. ч. для самостоятельной работы обучающихся)		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.3.1	Молчанова Е.И. Методические указания по изучению дисциплины Б1.О.26 Объектно-ориентированное программирование по направлению подготовки 09.03.04 Программная инженерия, профиль Разработка программно-информационных систем / Е.И. Молчанова ; ИрГУПС. – Иркутск : ИрГУПС, 2019. – 15 с. - Текст: электронный. - URL: https://www.irgups.ru/eis/for_site/umkd_files/mu_2425_1398_2020_1_signed.pdf	Онлайн
6.2 Ресурсы информационно-телекоммуникационной сети «Интернет»		
6.2.1	Электронно-библиотечная система «Издательство Лань», https://e.lanbook.com/	
6.2.2	Электронно-библиотечная система «Университетская библиотека онлайн», https://biblioclub.ru/	
6.3 Программное обеспечение и информационные справочные системы		
6.3.1 Базовое программное обеспечение		
6.3.1.1	Microsoft Windows Professional 10, государственный контракт от 20.07.2021 № 0334100010021000013-01	
6.3.1.2	Microsoft Office Russian 2010, государственный контракт от 20.07.2021 № 0334100010021000013-01	
6.3.1.3	FoxitReader, свободно распространяемое программное обеспечение http://free-software.com.ua/pdf-viewer/foxit-reader/	
6.3.1.4	Adobe Acrobat Reader DC свободно распространяемое программное обеспечение https://get.adobe.com/ru/reader/enterprise/	
6.3.1.5	Яндекс. Браузер. Прикладное программное обеспечение общего назначения, Офисные приложения, лицензия – свободно распространяемое программное обеспечение по лицензии BSD License	
6.3.2 Специализированное программное обеспечение		
6.3.2.1	NetBeans IDE, свободная лицензия Apache License 2.0 https://www.apache.org/licenses	

6.3.2.2	Java Virtual Machine, свободная лицензия Oracle Java SE https://www.oracle.com/downloads/licenses/javase-license1.html: Argo UML УЧ. ПРОЦ. http://argouml.ru.uptodown.com/windows
6.3.2.3	NetBeans IDE, свободная лицензия Apache License 2.0 https://www.apache.org/licenses/ Java Virtual Machine, свободная лицензия Oracle Java SE https://www.oracle.com/downloads/licenses/javase-license1.html
6.3.2.4	Argo UML УЧ. ПРОЦ. http://argouml.ru.uptodown.com/windows
6.3.2.5	NetBeans IDE, свободная лицензия Apache License 2.0 https://www.apache.org/licenses/ Java Virtual Machine, свободная лицензия Oracle Java SE https://www.oracle.com/downloads/licenses/javase-license1.html
6.3.3 Информационные справочные системы	
6.3.3.1	Не предусмотрены
6.4 Правовые и нормативные документы	
6.4.1	Не предусмотрены

7 ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ УЧЕБНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

1	Корпуса А, Б, В, Г, Д, Е ИрГУПС находятся по адресу г. Иркутск, ул. Чернышевского, д. 15; корпус Л ИрГУПС находится – по адресу г. Иркутск, ул. Лермонтова, д.80
2	Учебная аудитория Д-518* для проведения лекционных и практических занятий, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, ноутбук (переносной). Для проведения занятий имеются учебно-наглядные пособия (презентации).
3	Учебная аудитория Д-417*для проведения лекционных и практических занятий, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, ноутбук (переносной). Для проведения занятий имеются учебно-наглядные пособия (презентации).
4	Компьютерный класс «Информатика» Д-501 для проведения практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, компьютеры с подключением к сети Интернет, обеспечивающие доступ в электронную информационно-образовательную среду ИрГУПС, мультимедиапроектор (переносной), экран (переносной), компьютер. Для проведения занятий имеются учебно-наглядные пособия (презентации).
5	Компьютерный класс «Информатика». «Технологии и методы программирования» Д-503 для проведения практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, компьютеры с подключением к сети Интернет, обеспечивающие доступ в электронную информационно-образовательную среду ИрГУПС, мультимедиапроектор (переносной), экран (переносной), компьютер. Для проведения занятий имеются учебно-наглядные пособия (презентации).
6	Компьютерный класс А-513 для проведения практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор (переносной), экран (переносной), компьютеры с подключением к сети Интернет, обеспечивающие доступ в электронную информационно-образовательную среду ИрГУПС.
7	Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к информационно-телекоммуникационной сети «Интернет», и обеспечены доступом в электронную информационно-образовательную среду ИрГУПС. Помещения для самостоятельной работы обучающихся: – читальные залы; – учебные залы вычислительной техники А-401, А-509, А-513, А-516, Д-501, Д-503, Д-505, Д-507; – помещения для хранения и профилактического обслуживания учебного оборудования – А-521

8 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Вид учебной деятельности	Организация учебной деятельности обучающегося
Лекция	Лекция (от латинского «lection» – чтение) – вид аудиторных учебных занятий. Лекция: закладывает основы научных знаний в систематизированной, последовательной,

	обобщенной форме; раскрывает состояние и перспективы развития соответствующей области науки и техники; концентрирует внимание обучающихся на наиболее сложных, узловых вопросах; стимулирует познавательную активность обучающихся. Во время лекционных занятий обучающийся должен уметь сконцентрировать внимание на изучаемых проблемах и включить в работу все виды памяти: словесную, образную и моторно-двигательную. Для этого весь материал, излагаемый преподавателем, обучающемуся необходимо конспектировать. На полях конспекта следует помечать вопросы, выделенные обучающимся для консультации с преподавателем. Выводы, полученные в виде формул, рекомендуется в конспекте подчеркивать или обводить рамкой, чтобы лучше запоминались. Полезно составить краткий справочник, содержащий определения важнейших понятий лекции. К каждому занятию следует разобрать материал предыдущей лекции. Изучая материал по учебнику или конспекту лекций, следует переходить к следующему вопросу только в том случае, когда хорошо усвоен предыдущий вопрос. Ряд вопросов дисциплины может быть вынесен на самостоятельное изучение. Такое задание требует оперативного выполнения. В конспекте лекций необходимо оставить место для освещения упомянутых вопросов. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале, то необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии
Практическое занятие	Практическое занятие – вид аудиторных учебных занятий, целенаправленная форма организации учебного процесса, при реализации которой обучающиеся по заданию и под руководством преподавателя выполняют практические задания. Практические задания направлены на углубление научно-теоретических знаний и овладение определенными методами работы, в процессе которых вырабатываются умения и навыки выполнения тех или иных учебных действий в данной сфере науки. Практические занятия развивают научное мышление и речь, позволяют проверить знания обучающихся, выступают как средства оперативной обратной связи; цель практических занятий – углублять, расширять, детализировать знания, полученные на лекции, в обобщенной форме и содействовать выработке навыков профессиональной деятельности. На практических занятиях подробно рассматриваются основные вопросы дисциплины, разбираются основные типы задач. К каждому практическому занятию следует заранее самостоятельно выполнить домашнее задание и выучить лекционный материал к следующей теме. Систематическое выполнение домашних заданий обязательно и является важным фактором, способствующим успешному усвоению дисциплины
Лабораторная работа	Основной целью лабораторных работ является теоретическое обоснование, наглядное и/или экспериментальное подтверждение и/или проверка существенных теоретических положений (законов, закономерностей) анализ существующих методик и методов их реализации и т.д. Они занимают преимущественное место при изучении дисциплин обязательной части и части, формируемой участниками образовательных отношений Блока 1. Исходя из цели, содержанием лабораторных работ могут быть: - экспериментальная проверка формул, методик расчета; - проведение натурных измерений свойств, рабочих параметров, режимов работы при помощи лабораторного оборудования и/или стендов и макетов; - ознакомление, анализ и теоретические выкладки по устройству, принципу действия и способам обслуживания аппаратов, деталей машин, механизмов, процессов, протекающих в них при этом и т.д.; - наглядная графическая интерпретация чертежей, схем, объемных поверхностей и т.д., воспроизводимых с помощью специализированного программного обеспечения; - имитационное моделирование процессов, протекающих в сложных химических, физических, механических, электрических и пр. объектах; - наглядное представление о работе персонала конкретной организации или подразделения ОАО «РЖД» посредством моделирования штатных и внештатных ситуаций в виртуальных специализированных АРМ (автоматизированных рабочих мест); - установление и подтверждение закономерностей (путем сравнения проведенного эксперимента и рассчитанных значений) и т.д.; - ознакомление с методиками проведения экспериментов, наглядным устройством стенд-макетов и пр.; - установление свойств веществ, их качественных и количественных характеристик; - анализ различных характеристик процессов, в том числе производственных и иных процессов; - расчет параметров различных явлений и процессов, смоделировать которые не возможно в реальных условиях (например, чрезвычайные ситуации и пр.); - наблюдение развития явлений, процессов и др.

	Допускается иное содержание лабораторных работ, если это будет способствовать реализации целей и задач дисциплины и формированию соответствующих компетенций. По характеру выполняемых лабораторных работ возможны: - ознакомительные работы, используемые для закрепления изученного теоретического материалы; - аналитические работы, используемые для получения новой информации на основе формализованных методов; - творческие работы, ориентированные на самостоятельный выбор подходов решения задач. Прежде, чем приступить к лабораторным занятиям, обучающимся необходимо повторить теоретический материал по теме работы. Каждая лабораторная работа оснащена методическими указаниями, разработанными преподавателями, ведущими дисциплину
Самостоятельная работа	Обучение по дисциплине «Объектно-ориентированное программирование» предусматривает активную самостоятельную работу обучающегося. В разделе 4 рабочей программы, который называется «Структура и содержание дисциплины», все часы самостоятельной работы расписаны по темам и вопросам, а также указана необходимая учебная литература: обучающийся изучает учебный материал, разбирает примеры и решает разноуровневые задачи в рамках выполнения как общих домашних заданий, так и индивидуальных домашних заданий (ИДЗ) и других видов работ, предусмотренных рабочей программой дисциплины. При выполнении домашних заданий обучающемуся следует обратиться к задачам, решенным на предыдущих практических занятиях, решенным домашним работам, а также к примерам, приводимым лектором. Если этого будет недостаточно для выполнения всей работы можно дополнительно воспользоваться учебными пособиями, приведенными в разделе 6.1 «Учебная литература». Если, несмотря на изученный материал, задание выполнить не удается, то в обязательном порядке необходимо посетить консультацию преподавателя, ведущего практические занятия, и/или консультацию лектора. Домашние задания, индивидуальные домашние задания и другие работы, предусмотренные рабочей программой дисциплины должны быть выполнены обучающимся в установленные преподавателем сроки в соответствии с требованиями к оформлению текстовой и графической документации, сформулированным в Положении «Требования к оформлению текстовой и графической документации. Нормоконтроль»
Комплекс учебно-методических материалов по всем видам учебной деятельности, предусмотренным рабочей программой дисциплины (модуля), размещен в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет	

Приложение № 1 к рабочей программе

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля успеваемости и промежуточной аттестации

1. Общие положения

Фонд оценочных средств (ФОС) является составной частью нормативно-методического обеспечения системы оценки качества освоения обучающимися образовательной программы.

Фонд оценочных средств предназначен для использования обучающимися, преподавателями, администрацией ИрГУПС, а также сторонними образовательными организациями для оценивания качества освоения образовательной программы и уровня сформированности компетенций у обучающихся.

Задачами ФОС являются:

– оценка достижений обучающихся в процессе изучения дисциплины;

– обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения в образовательный процесс;

– самоподготовка и самоконтроль обучающихся в процессе обучения.

Фонд оценочных средств сформирован на основе ключевых принципов оценивания: валидность, надежность, объективность, эффективность.

Для оценки уровня сформированности компетенций используется трехуровневая система:

– минимальный уровень освоения, обязательный для всех обучающихся по завершению освоения образовательной программы; дает общее представление о виде деятельности, основных закономерностях функционирования объектов профессиональной деятельности, методов и алгоритмов решения практических задач;

– базовый уровень освоения, превышение минимальных характеристик сформированности компетенций; позволяет решать типовые задачи, принимать профессиональные и управленческие решения по известным алгоритмам, правилам и методикам;

– высокий уровень освоения, максимально возможная выраженность характеристик компетенций; предполагает готовность решать практические задачи повышенной сложности, нетиповые задачи, принимать профессиональные и управленческие решения в условиях неполной определенности, при недостаточном документальном, нормативном и методическом обеспечении.

2. Перечень компетенций, в формировании которых участвует дисциплина.
Программа контрольно-оценочных мероприятий.
Показатели оценивания компетенций, критерии оценки

Дисциплина «Объектно-ориентированное программирование» участвует в формировании компетенций:

ОПК-2. Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности

ОПК-5. Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем

ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов

Программа контрольно-оценочных мероприятий очная форма обучения

№	Наименование контрольно-оценочного мероприятия	Объект контроля	Код индикатора достижения компетенции	Наименование оценочного средства (форма проведения*)
2 семестр				
1.0	**Раздел 1. Основы языка программирования Java. Программы. Данные. Организация данных**			
1.1	Текущий контроль	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 1	ОПК-2.1 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно) Тестирование (компьютерные технологии)
1.2	Текущий контроль	Тема 1: Введение. Технологии Java SE, JEE. История и место Java среди современных языков программирования. Основные термины и инструментарий разработчика. Запуск приложения в командной строке. Передача параметров через командную строку	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно)
1.3	Текущий контроль	Тема 2: Знакомство со средой программирования NetBeans IDE. Установка среды. Структура кода простейшего приложения Java. Определение и вызов метода. Компиляция файлов проекта и запуск приложения. Структура проекта NetBeans. Отладка	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
1.4	Текущий контроль	Тема 3: Общие представления о языке Java. Алфавит Константы. Специальные символы. Управляющие конструкции. Составной оператор. Условный оператор if . Оператор выбора switch. Условное выражение …?... : Особенность операторов инкремента ++ и декремента Оператор цикла с пареметром for. Оператор цикла с предусловием while. Оператор цикла с постусловием do...while.	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)

1.5	Текущий контроль	Тема 4: Оперативное хранение данных в языке Java. Примитивные типы переменных. Приведение числовых типов. Ссылочные переменные	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно)
1.6	Текущий контроль	Тема 5: Ссылочные переменные и объекты. Процедурное и объектно-ориентированное программирование. Классы, объекты и объектные переменные. Классы для типов данных и структур данных	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно)
1.7	Текущий контроль	Тема 6: Массивы как объекты. Описание. Длина массива. Многомерные массивы. Зубчатый массив. Сравнение и копирование массивов. Класс Arrays. Оператор цикла for для перебора элементов массива (foreach)	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
1.8	Текущий контроль	Тема 7: Работа со строками Классы для работы со строками String, StringBuffer, StringBuilder. Создание строк и конкатенация. Методы для манипулирования строками. Консольный ввод. Форматированный вывод	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
2.0	**Раздел 2. Объектная модель языка Java**			
2.1	Текущий контроль	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 2	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно) Тестирование (компьютерные технологии)
2.2	Текущий контроль	Тема 8: Введение в объектно-ориентированное программирование. Объектно-ориентированная парадигма программирования. Основные понятия объектного подхода	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тестирование (компьютерные технологии)
2.3	Текущий контроль	Тема 9: Объектная модель языка Java. Объявление классов. Заголовок класса. Модификаторы. Пакеты. Ключевое слово class. Ключевое слово extends. Ключевое слово implements. Тело класса. Объявление полей. Объявление методов. Объявление конструкторов. Объектные инициализаторы. Перегрузка (overload). Методы с переменным числом параметров	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно) Тестирование (компьютерные технологии)

2.4	Текущий контроль	Тема 10: Базовые принципы ОО подхода. Абстракция и инкапсуляция. Наследование и полиморфизм. Использование super и this. Переопределение методов. Абстрактные классы. Интерфейсы. Клонирование объектов	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно) Тестирование (компьютерные технологии)
2.5	Текущий контроль	Тема 11: Ошибки при работе программы. Исключения (Exceptions). Конструкция try-catch-finally. Использование операторов throw, throws. Классы исключений. Создание пользовательских классов исключений	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
	Промежуточная аттестация	Раздел 1. Основы языка программирования Java. Программы. Данные. Организация данных Раздел 2. Объектная модель языка Java	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Зачет (собеседование) Зачет - тестирование (компьютерные технологии)
3 семестр				
3.0	**Раздел 3. Введение в проектирование классов**			
3.1	Текущий контроль	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 3	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно) Тестирование (компьютерные технологии)
3.2	Текущий контроль	Тема 12: Объектно-ориентированный анализ. Язык UML.Рекомендации по проектированию классов. Класс Object	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
3.3	Текущий контроль	Тема 13: Вложенные типы. Внутренние, вложенные, анонимные классы	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
4.0	**Раздел 4. Использование классов и библиотек**			
4.1	Текущий контроль	Проработка лекционного материала и подготовка к лабораторным занятиям раздела 4	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2	Собеседование (устно) Тестирование (компьютерные технологии)

			ОПК-6.3	
4.2	Текущий контроль	Тема 14. Потоки выполнения. Класс Thread и интерфейс Runnable. Потоки-демоны	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
4.3	Текущий контроль	Тема 15. Потоки ввода/вывода. Байтовые и символьные потоки. Класс File. Вывод/Ввод текстовой информации. Класс RandomAccessFile. Сериализация объектов Класс Scanner	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
4.4	Текущий контроль	Тема 16. Параметризация типов и ее применение. Обобщения (Generic). Параметризированные типы. Метасимвол «?». Ограничения. Параметризованные методы	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
4.5	Текущий контроль	Тема 17. Каркас коллекций. Очереди, стеки, списки, множества. Карты отображений	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
4.6	Текущий контроль	Тема 18. Введение в программирование GUI средствами Swing. Связь с архитектурой MVC. Размещение компонентов. Обработка событий. Рисование средствами Swing	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
4.7	Текущий контроль	Тема 19. JDBC. Драйверы, соединения и запросы	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
	Промежуточная аттестация	Раздел 3. Введение в проектирование классов. Раздел 4. Использование классов и библиотек	ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Экзамен (собеседование) Экзамен - тестирование (компьютерные технологии)

*Форма проведения контрольно-оценочного мероприятия: устно, письменно, компьютерные технологии.

Описание показателей и критериев оценивания компетенций.
Описание шкал оценивания

Контроль качества освоения дисциплины включает в себя текущий контроль успеваемости и промежуточную аттестацию. Текущий контроль успеваемости и промежуточная аттестация обучающихся проводятся в целях установления соответствия достижений обучающихся поэтапным требованиям образовательной программы к результатам обучения и формирования компетенций.

Текущий контроль успеваемости – основной вид систематической проверки знаний, умений, навыков обучающихся. Задача текущего контроля – оперативное и регулярное управление учебной деятельностью обучающихся на основе обратной связи и корректировки. Результаты оценивания учитываются в виде средней оценки при проведении промежуточной аттестации.

Для оценивания результатов обучения используется четырехбалльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» и двухбалльная шкала: «зачтено», «не зачтено».

Перечень оценочных средств, используемых для оценивания компетенций, а также краткая характеристика этих средств приведены в таблице.

Текущий контроль

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Собеседование	Средство контроля на практическом занятии, организованное как специальная беседа преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п. Может быть использовано для оценки знаний обучающихся	Вопросы для собеседования по темам/разделам дисциплины
2	Тестирование (компьютерные технологии)	Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Фонд тестовых заданий
3	Лабораторная работа	Средство, позволяющее оценить умение обучающегося письменно/устно излагать суть поставленной задачи, самостоятельно применять стандартные методы решения поставленной задачи с использованием имеющейся лабораторной базы, проводить анализ полученного результата работы. Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся	Образец задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

Промежуточная аттестация

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Зачет	Средство, позволяющее оценить знания, умения, навыков и (или) опыта деятельности обучающегося по дисциплине. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Перечень теоретических вопросов и практических заданий к зачету
2	Тест – промежуточная аттестация в форме зачета	Система автоматизированного контроля освоения компетенций (части компетенций) обучающимся по	Фонд тестовых заданий

		дисциплине (модулю) с использованием информационно-коммуникационных технологий. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	
3	Экзамен	Средство, позволяющее оценить знания, умения, навыков и (или) опыта деятельности обучающегося по дисциплине. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Перечень теоретических вопросов и практических заданий (образец экзаменационного билета) к экзамену
4	Тест – промежуточная аттестация в форме экзамена	Система автоматизированного контроля освоения компетенций (части компетенций) обучающимся по дисциплине (модулю) с использованием информационно-коммуникационных технологий. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Фонд тестовых заданий

Критерии и шкалы оценивания компетенций в результате изучения дисциплины при проведении промежуточной аттестации в форме зачета и экзамена. Шкала оценивания уровня освоения компетенций

Шкалы оценивания		Критерии оценивания	Уровень освоения компетенции
«отлично»	«зачтено»	Обучающийся правильно ответил на теоретические вопросы. Показал отличные знания в рамках учебного материала. Правильно выполнил практические задания. Показал отличные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на все дополнительные вопросы	Высокий
«хорошо»		Обучающийся с небольшими неточностями ответил на теоретические вопросы. Показал хорошие знания в рамках учебного материала. С небольшими неточностями выполнил практические задания. Показал хорошие умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на большинство дополнительных вопросов	Базовый
«удовлетворительно»		Обучающийся с существенными неточностями ответил на теоретические вопросы. Показал удовлетворительные знания в рамках учебного материала. С существенными неточностями выполнил практические задания. Показал удовлетворительные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Допустил много неточностей при ответе на дополнительные вопросы	Минимальный
«неудовлетворительно»	«не зачтено»	Обучающийся при ответе на теоретические вопросы и при выполнении практических заданий продемонстрировал недостаточный уровень знаний и умений при решении задач в рамках учебного материала. При ответах на дополнительные вопросы было допущено множество неправильных ответов	Компетенция не сформирована

Тест – промежуточная аттестация в форме зачета и экзамена

Шкала оценивания		Критерии оценивания
«отлично»	«зачтено»	Обучающийся верно ответил на 90 – 100 % тестовых заданий при прохождении тестирования
«хорошо»		Обучающийся верно ответил на 80 – 89 % тестовых заданий при прохождении тестирования
«удовлетворительно»		Обучающийся верно ответил на 70 – 79 % тестовых заданий при прохождении тестирования
«неудовлетворительно»	«не зачтено»	Обучающийся верно ответил на 69 % и менее тестовых заданий при прохождении тестирования

Критерии и шкалы оценивания результатов обучения при проведении текущего контроля успеваемости

Собеседование

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Глубокое и прочное усвоение программного материала. Полные, последовательные, грамотные и логически излагаемые ответы при видоизменении задания. Обучающийся свободно справляется с поставленными задачами, может обосновать принятые решения, демонстрирует владение разносторонними навыками и приемами выполнения практических работ
«хорошо»		Знание программного материала, грамотное изложение, без существенных неточностей в ответе на вопрос, правильное применение теоретических знаний, владение необходимыми навыками при выполнении практических задач
«удовлетворительно»		Обучающийся демонстрирует усвоение основного материала, при ответе допускаются неточности, при ответе недостаточно правильные формулировки, нарушение последовательности в изложении программного материала, затруднения в выполнении практических заданий Слабое знание программного материала, при ответе возникают ошибки, затруднения при выполнении практических работ
«неудовлетворительно»	«не зачтено»	Не было попытки выполнить задание

Тестирование

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Обучающийся верно ответил на 90 – 100 % тестовых заданий при прохождении тестирования
«хорошо»		Обучающийся верно ответил на 80 – 89 % тестовых заданий при прохождении тестирования
«удовлетворительно»		Обучающийся верно ответил на 70 – 79 % тестовых заданий при прохождении тестирования
«неудовлетворительно»	«не зачтено»	Обучающийся верно ответил на 69 % и менее тестовых заданий при прохождении тестирования

Лабораторная работа

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет без замечаний. Лабораторная работа выполнена обучающимся в полном объеме с соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; показал необходимые для проведения работы теоретические знания, практические умения и навыки. Работа (отчет) оформлена аккуратно, в наиболее оптимальной для фиксации результатов форме
«хорошо»		Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет с небольшими недочетами. Лабораторная работа выполнена обучающимся в полном объеме и самостоятельно. Допущены отклонения от необходимой

		последовательности выполнения, не влияющие на правильность конечного результата. Работа показывает знание обучающимся основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допущены неточности и небрежность в оформлении результатов работы (отчета)
«удовлетворительно»		Лабораторная работа выполнена с задержкой, письменный отчет с недочетами. Лабораторная работа выполняется и оформляется обучающимся при посторонней помощи. На выполнение работы затрачивается много времени. Обучающийся показывает знания теоретического материала, но испытывает затруднение при самостоятельной работе с источниками знаний или приборами
«неудовлетворительно»	«не зачтено»	Лабораторная работа не выполнена, письменный отчет не представлен. Результаты, полученные обучающимся, не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Показывается плохое знание теоретического материала и отсутствие необходимых умений. Лабораторная работа не выполнена, у учащегося отсутствуют необходимые для проведения работы теоретические знания, практические умения и навыки

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

3.1 Типовые контрольные задания для проведения собеседования

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для проведения собеседований.

Образец типового варианта вопросов для проведения собеседования
«Проработка лекционного материала и подготовка к лабораторным занятиям раздела 1»

1. Являются ли допустимыми и различными идентификаторы myObject, MyObject, myobject, Myobject, my object, my_object?
2. С каких символов начинается многострочный комментарий, предназначенный для автоматического создания документации по классу?
3. Включаются ли в программу метаданные при ее компиляции?
4. Каким символом заканчивается тело метода?
5. Какой тип подпрограмм (процедуры или функции) используется в языке Java?

Образец типового варианта вопросов для проведения собеседования
«Проработка лекционного материала и подготовка к лабораторным занятиям раздела 2»

1. Базовые принципы ОО подхода.
2. Объявление классов. Заголовок класса.
3. Модификаторы класса. Разграничение доступа в Java.
4. Тело класса (компоненты).

Образец типового варианта вопросов для проведения собеседования
«Проработка лекционного материала и подготовка к лабораторным занятиям раздела 3»

1. Размещение компонентов GUI Swing. Связь с архитектурой MVC. Обработка событий.
2. Параметризация типов и ее применение.
3. Потоки выполнения. Класс Thread и интерфейс Runnable. Жизненный цикл потока.
4. Потоки-демоны. Методы synchronized.

Образец типового варианта вопросов для проведения собеседования
«Проработка лекционного материала и подготовка к лабораторным занятиям раздела 4»

1. Размещение компонентов GUI Swing. Связь с архитектурой MVC. Обработка событий.
2. Параметризация типов и ее применение.
3. Потоки выполнения. Класс Tpread и интерфейс Runnable. Жизненный цикл потока.
4. Потоки-демоны. Методы synchronized.

3.2 Типовые контрольные задания для проведения тестирования

Фонд тестовых заданий по дисциплине содержит тестовые задания, распределенные по разделам и темам, с указанием их количества и типа.

Структура фонда тестовых заданий по дисциплине

Индикатор достижения компетенции	Тема в соответствии с РПД/РПП	Характеристика ТЗ	Количество тестовых заданий, типы ТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 1: Введение. Технологии Java SE, JEE. История и место Java среди современных языков программирования. Основные термины и инструментарий разработчика. Запуск приложения в командной строке. Передача параметров через командную строку	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 3 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 2: Знакомство со средой программирования NetBeans IDE. Установка среды. Структура кода простейшего приложения Java. Определение и вызов метода. Компиляция файлов проекта и запуск приложения. Структура проекта NetBeans. Отладка	Знание	3 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 3: Общие представления о языке Java. Алфавит Константы. Специальные символы. Управляющие конструкции. Составной оператор. Условный оператор if . Оператор выбора switch. Условное выражение …?... : Особенность операторов инкремента ++ и декремента Оператор цикла с пареметром for. Оператор цикла с предусловием while. Оператор цикла с постусловием do...while	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 3 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 4: Оперативное хранение данных в языке Java. Примитивные типы переменных. Приведение числовых типов. Ссылочные переменные	Знание	3– ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2	Тема 5: Ссылочные переменные и объекты. Процедурное и объектно-ориентированное программирование. Классы, объекты и объектные переменные. Классы для типов данных и структур данных	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 3 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ

ОПК-6.3			
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.2 ОПК-6.3	Тема 6: Массивы как объекты. Описание. Длина массива. Многомерные массивы. Зубчатый массив. Сравнение и копирование массивов. Класс Arrays. Оператор цикла for для перебора элементов массива (foreach)	Знание	3 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 7: Работа со строками Классы для работы со строками String, StringBuffer, StringBuilder. Создание строк и конкатенация. Методы для манипулирования строками. Консольный ввод. Форматированный вывод	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 3 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 8: Введение в объектно-ориентированное программирование. Объектно-ориентированная парадигма программирования. Основные понятия объектного подхода	Знание	3– ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 9: Объектная модель языка Java. Объявление классов. Заголовок класса. Модификаторы. Пакеты. Ключевое слово class. Ключевое слово extends. Ключевое слово implements. Тело класса. Объявление полей. Объявление методов. Объявление конструкторов. Объектные инициализаторы. Перегрузка (overload). Методы с переменным числом параметров	Знание	3– ОТЗ 2 – ЗТЗ
		Умение	2 – ОТЗ 4 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 10: Базовые принципы ОО подхода. Абстракция и инкапсуляция. Наследование и полиморфизм. Использование super и this. Переопределение методов. Абстрактные классы. Интерфейсы. Клонирование объектов	Знание	3 – ОТЗ 1 – ЗТЗ
		Умение	1– ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 11: Ошибки при работе программы. Исключения (Exceptions). Конструкция try-catch-finally. Использование операторов throw, throws. Классы исключений. Создание пользовательских классов исключений	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 12: Объектно-ориентированный анализ. Язык UML.Рекомендации по проектированию классов. Класс Object	Знание	2– ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1	Тема 13: Вложенные типы. Внутренние, вложенные, анонимные классы	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ

ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 14. Потоки выполнения. Класс Thread и интерфейс Runnable. Потоки-демоны	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2– ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 15. Потоки ввода/вывода. Байтовые и символьные потоки. Класс File. Вывод/Ввод текстовой информации. Класс RandomAccessFile. Сериализация объектов Класс Scanner	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Действие	1– ОТЗ 2 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 16. Параметризация типов и ее применение. Обобщения (Generic). Параметризированные типы. Метасимвол «?»..Ограничения. Параметризованные методы	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 17. Каркас коллекций. Очереди, стеки, списки, множества. Карты отображений	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1– ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 18. Введение в программирование GUI средствами Swing. Связь с архитектурой MVC. Размещение компонентов. Обработка событий. Рисование средствами Swing	Знание	2– ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-5.1 ОПК-5.2 ОПК-5.3 ОПК-6.1 ОПК-6.2 ОПК-6.3	Тема 19. JDBC. Драйверы, соединения и запросы	Знание	2 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 2 – ЗТЗ
		Действие	1– ОТЗ 1 – ЗТЗ
		Итого	81 – ОТЗ 81 – ЗТЗ

Полный комплект ФТЗ хранится в электронной информационно-образовательной среде ИрГУПС и обучающийся имеет возможность ознакомиться с демонстрационным вариантом ФТЗ.

Ниже приведен образец типового варианта итогового теста, предусмотренного рабочей программой дисциплины.

1. Установите соответствие терминов

. #	Вопрос	Ответ
a)	Управляющий класс – класс, отвечающий за координацию действий других классов	**control class**
b)	Класс – сущность – пассивный класс, информация о котором должна храниться постоянно и не уничтожаться с выключением системы	**entity class**
c)	Граничный класс – класс, который располагается на границе системы с внешней средой и непосредственно взаимодействует с актерами	**boundary class**
d)		environment class
e)		manager class
f)		border class

2. Укажите результат:

```
public class Dragon {
static String type = “Just Dragon";
public static void main(String [] s){
String type = “Black Dragon";
System.out.println(type); }
}
```

a) //выведет “Just Dragon"
b) //выведет “Black Dragon"

3. Какое описание класса содержит синтаксическую ошибку? Код написан в файле Quest1.java

a) public class Quest1 {}
b) public static class Quest1 {}
c) public abstract final class Quest1 {}
d) private class Quest1 {}
e) final class Quest1 {}

4. Объявление класса состоит из

a) заголовка и тела класса
b) заголовка класса
c) сигнатуры класса

5. Отметьте два верных утверждения. Модификатор поля static означает, что:

a) **поле принадлежит структуре класса**
b) **одно значение присуще всем экземплярам**
c) значение поля нельзя изменять
d) поле не может быть использовано нигде, кроме данного класса или его экземпляра

6. В объявление метода-конструктора не входит:
a) модификаторы
b) тип возвращаемого значения
c) имя класса
d) количество аргументов
e) тип аргументов;
f) порядок следования аргументов

7. В сигнатуру метода не входит:
a) тип возвращаемого значения
b) имя метода
c) количество аргументов
d) тип аргументов
e) порядок следования аргументов

8. Отметьте корректные пары строк кода (две):
class Point {

a)	**void get() {}** **void get(int x) {}**
b)	void get() {} int get() {}
c)	**void get(int x, double y) {}** **void get(double x, int y) {}**
d)	void get(int x) {} void get(int y) {}
e)	public int get() {} private int get() {}
	}

9. Ключевое слово This (укажите неверное утверждение):
a) возвращает ссылку на данный объект; позволяет передать аргумент, равный ссылке
b) на данный объект, в другой метод класса реализует "затеняющие" объявления
c) служит для обращения к полям, которые объявляются ниже
d) используется в конструкторах для явного вызова в первой строке другого конструктора этого же класса
e) **необходимо для обращения к полям класса**

10. Сколько конструкторов вызовется при создании объекта в строке 1. Дан код:

```
public class Quest5 {
public Quest5 () {}
public Quest5 (int i)
{this (i, i);}
public Quest5 (int i, int j)
{this ();}
public static void main (String [] args) {
Quest5 q = new Quest5 (2,3);     //1
}
}
```

Ответ: 2

11. Что выведется на консоль после компиляции и запуска этой программы?
Дан код:

```
public class Quest6 {
  public void meth (Number obj) {System.out.print ("1");}
```

```
  public void meth (Character obj) {System.out.print ("2");}
  private static void meth (Integer obj) {System.out.print ("3");}
  public void meth (int i) {System.out.print ("4");}
  public void meth (double d) {System.out.print ("5");}
  public static void main (String [] args) {
    Quest6 q = new Quest6 ();
    Number n = 67;
    Integer i = 78;
    q.meth (n);
   q.meth (i);
 }
}
```

Ответ: 13

12. Суперклассом для всех объектов является _____?

Ответ: Object

13. Обращение к родительскому конструктору записывается с помощью ключевого слова _____?

Ответ: super

14. Укажите слово - синоним для терминов атрибут, свойство _____?

Ответ: поле

15. Введите номер строки, в которая вызовет ошибку компиляции____?

Дан код:

```
public class Quest3 {
  public static int method () {
    final int loc;
  System.out.println (loc);//   1
  loc=4;//   2
    return loc+1;//   3
  }
  public static void main (String [] args) {
  method (); method (); method ();
  System.out.println (method ());
  }
}
```

Ответ: 1

16. При создании объекта автоматически вызывается _____?

Ответ: конструктор

17. Назовите метод, позволяющий получить уникальный целый номер для данного объекта _____

Ответ: hashCode()

18. Назовите метод, конвертирующий хэш-код в строку _____

Ответ: toString()

3.3 Типовые задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты.

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 2: Знакомство со средой программирования NetBeans IDE. Установка среды. Структура кода простейшего приложения Java. Определение и вызов метода. Компиляция файлов проекта и запуск приложения. Структура проекта NetBeans. Отладка»

1. Написать программу, получающую на вход в качестве аргумента два параметра - числа x и y. Большее из них умножить на 5. Вывести результат на экран.

2. Дано a,b,. Найти наибольшее **max** и наименьшее **min** . Найти d:

$$d = \frac{5 \cdot \max(a, b+2) - 4 \cdot \min(1-a, b)}{3 + \frac{\max(a,b+2)}{\min(1-a,b)}}$$

Контрольные вопросы

1) Объясните основное назначение метода main().
2) Расскажите о входном параметре метода main() String[] args.
3) В чем состоит разница между JDK (Java Development Kit) и JRE (Java Runtime Environment)?
4) Что означает ключевое слово void?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 3: Общие представления о языке Java. Алфавит Константы. Специальные символы. Управляющие конструкции. Составной оператор. Условный оператор if . Оператор выбора switch. Условное выражение …?... : Особенность операторов инкремента ++ и декремента Оператор цикла с пареметром for. Оператор цикла с предусловием while. Оператор цикла с постусловием do...while.»

1.Наиболее рационально вычислить Y и F . Использовать составной оператор.

$$Y = \begin{cases} x^3 + 1 & \text{если } x <= -3 \\ x(1+2^x) & \text{если } -3 < x <= 4 \\ \operatorname{tg} x & \text{если } x > 4 \end{cases} \qquad F = \begin{cases} e^{\sin x} & \text{если } x <= -3 \\ x^4 & \text{если } -3 < x <= 4 \\ \sqrt[5]{tgx} & \text{если } x > 4 \end{cases}$$

Контрольные вопросы

1. В чем отличие конструкций System.out.println() и System.out.print()?
2. Каким образом осуществляется конкатенация строк в Java?
3. Что означает ключевое слово static?
4. Какие три вида комментариев Вы знаете?
5. Как сравнивают объекты?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 6: Массивы как объекты. Описание. Длина массива. Многомерные массивы. Зубчатый массив. Сравнение и копирование массивов. Класс Arrays. Оператор цикла for для перебора элементов массива (foreach)»

Определить одномерный массив и заполнить его случайными значениями.

1. Все элементы массива, меньшие заданного значения, и их номера записать в новые массивы. Вывести новые массивы на экран. Заданное значение определяется константой.
2. Поменять местами максимальный и минимальный элементы массива. Вывести измененный массив на экран.

Контрольные вопросы

1. Назовите символы Escape-последовательности отвечающие за перевод строки и за табуляцию.
2. В каком интервале возвращает значение метод Math.random() и какой метод отвечает за округление числа?
3. Чем отличаются управляющие конструкции while, do while, for ?
4. Как объявляется и инициализируется одномерный массив.
5. Передача массива в метод.
6. Возврат массива из метода.

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 7: Работа со строками Классы для работы со строками String, StringBuffer, StringBuilder. Создание строк и конкатенация. Методы для манипулирования строками. Консольный ввод. Форматированный вывод»

1. Из небольшого текста удалить все символы, кроме пробелов, не являющиеся буквами. Между последовательностями подряд идущих букв оставить хотя бы один пробел.
2. Написать программу, преобразующую все прописные русские буквы строки S в строчные (остальные символы строки S не изменяются). Строка S является входным и выходным параметром.

Контрольные вопросы

1. Классы для работы со строками.
2. Способы создания переменной типа String.
3. Создание массива строк.
4. Способы конкатенации строк.
5. Наиболее употребительные методы класса String

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 9: Объектная модель языка Java. Объявление классов. Заголовок класса. Модификаторы. Пакеты. Ключевое слово class. Ключевое слово extends. Ключевое слово implements. Тело класса. Объявление полей. Объявление методов. Объявление конструкторов. Объектные инициализаторы. Перегрузка (overload). Методы с переменным числом параметров»

1. Создать программу на языке Java для определения класса в некоторой предметной области. Описать свойства, конструктор, методы **геттеры/сеттеры**, перекрыть метод **toString**() для вывода полной информации об объекте в отформатированном виде. Значения полей по умолчанию, выделенных фигурными скобками, задать в конструкторе, остальные – в объектном инициализаторе В соответствии с примером 1.6, создать программу на языке Java

для определения класса в некоторой предметной области. Описать свойства, конструктор, методы **геттеры/сеттеры**, перекрыть метод **toString**() для вывода полной информации об объекте в отформатированном виде. Значения полей по умолчанию, выделенных фигурными скобками, задать в конструкторе, остальные – в объектном инициализаторе.

Вариант 1). Записная книжка контактов.

Contact – запись информации о контакте в записную книжку.

Свойства:

- Id – идентификатор контакта;
- first-Name – имя;
- lastName –фамилия; address – адрес;

} Конструктор

- phone –телефон;
- note – запись о контакте.

2. Разработать класс, набор методов (конструктор и минимум два метода) для программной модели заданного объекта. Описание объекта и его основных свойств приводится ниже.

Объект «вектор на плоскости» заданный в системе декартовых координат. Начало вектора расположено в начале координат. Операции определяются согласно обще принятых формул линейной (векторной) алгебры. Предусмотреть возможность операции присваивания, сравнения модулей, скалярного умножения и распечатки координат текущих значений. Конструктор должен позволить создавать объекты без и с начальной инициализацией.

3. Создать классы, спецификации которых приведены ниже. Определить конструкторы и методы setТип(), getТип(), toString(). Определить дополнительно методы в классе, создающем массив объектов. Задать критерий выбора данных и вывести эти данные на консоль. В каждом классе, обладающем информацией, должно быть объявлено несколько конструкторов.

1. Student: id, Фамилия, Имя, Отчество, Дата рождения, Адрес, Телефон, Факультет, Курс, Группа.

Создать массив объектов. Вывести:

a) список студентов заданного факультета;
b) списки студентов для каждого факультета и курса;
c) список студентов, родившихся после заданного года;
d) список учебной группы.

Контрольные вопросы

1) Что такое сигнатура метода?
2) Зачем нужен конструктор?
3) Дайте определение объекту?
4) Зачем необходимо ключевое слово new?
5) Зачем нужны комментарии?
6) Описать сигнатуру пользовательского конструктора и конструктора по умолчанию.
7) Зачем нужны операторы импорта?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 10: Базовые принципы ОО подхода. Абстракция и инкапсуляция. Наследование и полиморфизм. Использование super и this. Переопределение методов. Абстрактные классы. Интерфейсы. Клонирование объектов»

Реализовать предметную область. На оценку 4-5 модифицировать полиморфный код с использованием абстрактного класса и интерфейса (оба варианта), основываясь на материалах лекции. Все полиморфные переменные, соответствующие объектам-наследникам, должны храниться в векторе, как в примере программного кода из материалов лекции.

1 Записная книжка. Создать родительский класс «Событие» (дата, время) и дочерние классы:

«День рождения» (именинник, место проведения праздника и возраст);

«Встреча» (человек с которым назначена встреча и место встречи);

«Другое» (описание).

Реализовать класс для хранения списка событий с методом добавления события и методом печати списка событий.

Контрольные вопросы

1. Дайте определение базовых принципов ОО подхода (лекция).
2. Как наследуются поля, конструкторы, методы?
3. Как обратиться к методу предка?
4. Что такое переопределение и перегрузка методов?
5. Как запретить переопределение метода?
6. Чем абстрактный класс отличается от конкретного?
7. Назовите стереотипы классов.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Тема 11: Ошибки при работе программы. Исключения (Exceptions). Конструкция try-catch-finally. Использование операторов throw, throws. Классы исключений. Создание пользовательских классов исключений»

1. Изучить исключения в Java и их обработку. Выполнить приведенный ниже пример в среде NetBeans и прокомментировать полученные результаты.
2. Модифицировать и выполнить предложенный код для использования оператора throws.
3. В проекте с заданием по ЛР5 создать класс исключения для обработки ошибок по своему варианту и продемонстрировать его применение.

Контрольные вопросы

1. Что такое исключение?
2. Опишите иерархию классов исключений.
3. На какие типы можно разделить все исключения в Java?
4. Опишите конструкцию для обработки исключений.
5. Каким образом можно перехватывать все виды исключений?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 12: Объектно-ориентированный анализ. Язык UML.Рекомендации по проектированию классов. Класс Object»

1. Выполнить объектно-ориентированный анализ проблемной области на примере системы резервирования авиабилетов.

a) Выполнить объектно-ориентированный анализ проблемной области на примере системы резервирования авиабилетов. Идентифицировать объекты, которые участвуют в сценарии и которые могут стать программными артефактами в объектно-ориентированной системе, моделирующей сценарий. Разработать диаграмму классов. Различать внутреннюю и внешнюю составляющие объектов, четко отделяя данные, которые объекты содержат и инкапсулируют (или защищают от большей системы), от интерфейса или сообщений, которые объекты получают.

b) Идентифицировать акторов системы резервирования авиабилетов. Для этого перечислить внешние сущности, которые используют систему. Разработать диаграмму вариантов использования, чтобы показать какие действия должна выполнять разрабатываемая система. Дать характеристику варианта использования, которая подробно описывает предусловия, поток событий и постусловия варианта использования. Идентифицировать и описать альтернативные сценарии, которые могут быть созданы для одного варианта использования.

2. В соответствии с номером варианта, выполнить объектно-ориентированный анализ проблемной области. Создать консольное приложение, удовлетворяющее требованиям, приведенным в задании. Наследование применять только в тех заданиях, в которых это логически обосновано. Аргументировать принадлежность классу каждого создаваемого метода и корректно переопределить для каждого класса методы equals(), hashCode(), toString().

3. В соответствии с номером варианта, выполнить объектно-ориентированный анализ проблемной области. Создать консольное приложение, удовлетворяющее следующим требованиям: использовать возможности ООП: классы, наследование, полиморфизм, инкапсуляция.

Контрольные вопросы

1. Типы отношений между классами.
2. UML принципы моделирования.
3. Виды диаграмм UML.
4. Элементы диаграммы Use Case.

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 13: Вложенные типы. Внутренние, вложенные, анонимные классы»

По аналогии с представленным примером программного кода выполнить реализацию программы на языке Java в соответствии с индивидуальным заданием с документационными и послестрочными комментариями. Создать диаграмму классов.

Пример вариантов заданий

1. Создать класс Notepad с внутренним классом или классами, с помощью объектов которого могут храниться несколько записей на одну дату.

2. Создать класс Payment с внутренним классом, с помощью объектов которого можно сформировать покупку из нескольких товаров.

Контрольные вопросы

1. Вложенные типы, определение и терминология фирмы Sun.
2. Причины для использования вложенных классов.

3. Статические вложенные классы. Обращение к статическим вложенным классам.
4. Вложенные интерфейсы, реализация, обращение.
5. Внутренние классы, определение, тип.
6. Доступны ли поля и методы внешнего объекта элементам внутреннего класса?
7. Создание объекта внутреннего класса из управляющего класса.
8. Как записать ссылку на объект внешнего класса во внутреннем классе?
9. Локальные классы.
10. Анонимные классы

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 14. Потоки выполнения. Класс Thread и интерфейс Runnable. Потоки-демоны»

Рассмотрим пример апплета, в котором создаются два потока, соединённые каналом ввода-вывода. Один поток рисует фейерверк, другой формирует для него цвет

Рисованием фейерверка занимается объект-поток типа ZalpThr. Другой поток (типа ColorThr) случайным образом генерирует цвет очередной серии залпов. Для передачи сгенерированного цвета в объект-поток рисования фейерверка используются канальные потоки ввода-вывода, которые создаются в объекте-апплете.

Контрольные вопросы

1. Многопоточная архитектура
2. Базовые классы для работы с потоками

Класс Thread

Интерфейс Runnable

3. Работа с приоритетами
4. Демон-потоки

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 15. Потоки ввода/вывода. Байтовые и символьные потоки. Класс File. Вывод/Ввод текстовой информации. Класс RandomAccessFile. Сериализация объектов Класс Scanner»

1. Выполнить задания из варианта В ЛР1, сохраняя объекты приложения в одном или нескольких файлах с применением механизма сериализации. Объекты могут содержать поля, помеченные как static, а также transient. Для изменения информации и извлечения информации в файле создать специальный класс-коннектор с необходимыми для выполнения этих задач методами.

2. При выполнении следующих заданий для вывода результатов создавать новую директорию и файл средствами класса File.

a) Создать и заполнить файл случайными целыми числами. Отсортировать содержимое файла по возрастанию.

b) Прочитать текст Java-программы и все слова public в объявлении атрибутов и методов класса заменить на слово private.

Контрольные вопросы

1. Какие основные концепции работы с файлами Вы знаете?
2. Назовите несколько методов класса File.
3. Какие классы работают с двоичными файлами?
4. Какие классы работают с текстовыми файлами?
5. Что такое кодировка?
6. Какие виды кодировочных таблиц Вы знаете?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 16. Параметризация типов и ее применение. Обобщения (Generic).
Параметризированные типы. Метасимвол «?». Ограничения. Параметризованные методы»

Используя классы объектов, созданных по варианту в лабораторной работе № 1, модифицировать приложение, разработав Generic-класс, манипулирующий группами объектов (добавлять, удалять и т.д.). В качестве параметра класс должен принимать различные объекты (использовать ограничения на тип).

Контрольные вопросы

1. Назначение generic.
2. Что такое неопределенный тип? Приведите примеры его задания и использования.
3. Что такое ограничения на неопределенный тип? Когда их можно использовать приведите примеры.
4. Как можно задать ограничения на неопределённый тип?
5. Что такое методы generic? Зачем они применяются? Приведите примеры.

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 17. Каркас коллекций. Очереди, стеки, списки, множества. Карты отображений»

1. Ввести строки из файла, записать их в стек. Вывести строки в файл в обратном порядке.
2. Ввести число, занести его цифры в стек. Вывести число, у которого цифры идут в обратном порядке.
3. Сложить два многочлена заданной степени, если коэффициенты многочленов хранятся в объекте HashMap.
4. Создать стек из элементов каталога.

Задание 2

1. Записная книжка контактов. Реализовать сортировку по дате и по времени.
2. Система управления доставкой товара. Реализовать сортировку по дате и по времени.
3. Телепрограмма. Реализовать сортировку по наименованию передачи и по времени.

Контрольные вопросы

1. Внедрены ли в каркас коллекций механизмы интеграции стандартных массивов?
2. Назовите два средства прохода по коллекции.
3. Можно ли в коллекцию поместить элементарный тип – int, char, long без упаковки?
4. Чем очередь отличается от стека?
5. Какие классы и интерфейсы используются для создания очереди и стека?
6. Что такое классы отображений, приведите пример.
7. Чем отличаются интерфейсы Collection и List?
8. Чем отличаются интерфейсы Comparator и Comparable?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 18. Введение в программирование GUI средствами Swing. Связь с архитектурой MVC.
Размещение компонентов. Обработка событий. Рисование средствами Swing»

1. Изучить основные понятия и термины обработки событий по модели делегирования событий на Java.
2. Разобрать приведенные примеры **FontsList**, **LinesDraw, KeyCodes, Timer**, **DrawFig**.

3. Разработать программу. Основная задача – разработка упрощенной имитации поведения объектов (все последующие лабораторные работы будут расширять это задание). Объекты реализуются через наследование: абстрактный класс, интерфейс → наследники.

Рабочий цикл программы:

- запускается процесс симуляции, генерируются объекты классов согласно заданию;
- симуляция завершается, выводится статистическая информация.

Для решения задачи:

- Создать интерфейс IBehaviour, задающий поведение объекта;
- Создать иерархию классов, определяющие объекты по варианту и реализующие интерфейс IBehaviour.
- Создать класс Habitat, определяющий размер рабочей области и хранящий список объектов, с параметрами, заданными вариантом. Предусмотреть в классе метод Update, вызывающийся по таймеру и получающий на вход время, прошедшее от начала симуляции. В данном методе должны генерироваться новые объекты и помещаться в поле визуализации в случайном месте. Визуализация объекта – схематично, плюсом будет, если объект будет похож на оригинал;

Рабочее окно программы – область визуализации среды обитания объектов;

4. Симуляция должна запускаться по клавише **B** и останавливаться по клавише **E**. При остановке симуляции список уничтожается. Время симуляции должно отображаться текстом в области визуализации и скрываться/показываться по клавише **T**;
5. По завершению симуляции в поле визуализации должна выводиться информация о количестве и типе сгенерированных объектов, а также время симуляции. Текст должен быть форматирован, т.е. выводиться с использованием разных шрифтов и цветов.
6. Параметры симуляции задаются в классе Habitat.

Контрольные вопросы

1. Что такое GUI?
2. Какие графические библиотеки есть в Java?
3. Какой принцип работы компонентов AWT? Недостатки библиотеки.
4. Какой принцип работы компонентов Swing?
5. Что такое элементы управления и что такое контейнеры?
6. Какие классы элементов управления существуют?
7. Что необходимо сделать, чтобы добавить компонент в контейнер?
8. Как можно перехватить и обработать события, пришедшие от компонентов?

Образец заданий для выполнения лабораторных работ
и примерный перечень вопросов для их защиты
«Тема 19. JDBC. Драйверы, соединения и запросы»

Задание

a) Установить соединение с базой данных
b) Внести записи в базу данных.
c) Выполнить запрос к базе данных.

Контрольные вопросы

1. Что такое JDBC?
2. Как осуществляется соединение?
3. Что передается в стороке URL?
4. Что делает класс Statement?
5. Какие есть типы запросов и классы для их выполнения?
6. Для чего нужен класс ResultSet?
7. Как извлекаются данные из ResultSet?
8. Как связаны табличная модель (Table Model) и класс JTable?

3.4 Перечень теоретических вопросов к зачету

(для оценки знаний)

Раздел 1. Основы языка программирования Java. Программы. Данные. Организация данных

1.1. Обязательно ли использование оператора default в составном операторе switch?

1.2. Какой оператор отсутствует в программном коде:

```
switch(i/j){
case 1:
i=0;
case 2:
i=2;
case 10:
i=3;
j=j/10;
default:
i=4;
};
```

1.3. Правильно ли записано выражение

```
j=i>6?i+1:i-2;
```

1.4. Чему равно значение переменных j и i после выполнения команд:

```
i=0;
j=i++;
```

1.5. Может ли блок изменения счетчиков в цикле for содержать несколько выражений?

1.6. Какой метод используется для ввода данных в окно диалога?

1.7. Какой метод применяют для вывода в консоль данных?

1.8. Написать в шестнадцатеричном виде числа 0, 1, 8, 15, 16, 255, 256.

1.9. Дать ответ, являются ли допустимыми идентификаторами i1, i_1, 1i, i&1, i1234567891011, IJKLMN?

1.10. Являются ли допустимыми и различными идентификаторы myObject, MyObject, myobject, Myobject, my object, my_object?

1.11. С каких символов начинается многострочный комментарий, предназначенный для автоматического создания документации по классу?

1.12. Включаются ли в программу метаданные при ее компиляции?

1.13. Каким символом заканчивается тело метода?

1.14. Какой тип подпрограмм (процедуры или функции) используется в языке Java?

1.15. Дать ответ, что такое класс, объект, поле данных, метод.

1.16. Назовите компоненты проекта.

1.17. Приведите структуру компонентов приложения и соответствующих им папок.

1.18. Опишите, как на диаграмме классов изображаются поля и методы объекта.

1.19. Создать новый проект NetBeans как Java Application (MyFirstAppliction). Импортировать классы пакета swing.

1.20. Добавить в класс приложения общедоступное числовое поле x типа integer, инициализированное неким значением, и поле y того же типа, но без инициализации.

1.21. В методе main ввести с помощью диалоговой панели JOptionPane значение поля y.

1.22. Создать документационные комментарии для полей x и y.

1.23. Вызвать генерацию документации для проекта, просмотреть в ней созданные комментарии. Особенности работы с массивами.

1.24. Особенности работы со строками.

Раздел 2. Объектная модель языка Java

2.1. Парадигмы программирования. Определение.

2.2. Перечислите основные модели, подходы, приемы программирования.

2.3. Императивное и декларативное программирование.

2.4. Объектно-ориентированная парадигма программирования.

2.5. Объектный подход к разработке программного обеспечения (этапы).
2.6. Базовые принципы ОО подхода.
2.7. Объявление классов. Заголовок класса.
2.8. Модификаторы класса. Разграничение доступа в Java.
2.9. Тело класса (компоненты).
2.10. Объявление методов. Сигнатура. Модификаторы.
2.11. Объявление конструктора.
2.12. Алгоритм создания объекта.
2.13. Объектные инициализаторы.
2.14. Ключевое слово This.
2.15. Переопределение метода.
2.16. Ключевое слово abstract.
2.17. Интерфейсы.
2.18. Динамические библиотеки.
2.19. Статическое и динамическое связывание методов Java. Полиморфизм.
2.20. Ошибки при работе программы. Исключения (Exceptions).
2.21. Причины возникновения исключительных ситуаций.
2.22. Конструкция try-catch.
2.23. Конструкция try-catch-finally.
2.24. Использование оператора throw. Порядок генерирования исключений.
2.25. Проверяемые и непроверяемые исключения.
2.26. Создание пользовательских классов исключений.

3.5 Перечень типовых простых практических заданий к зачету

(для оценки умений)

1 Дан код:

```
public class Quest6 {
public static void main (String [] args) {
System.out.print ("A");
main ("java7");
 }
private static void main (String args) {
System.out.print ("B");
}
}
```

Что будет выведено в результате запуска и компиляции?

a. ошибка компиляции
b. BA
c. AB
d. AA
e. компиляция пройдет успешно, а при выполнении программа зациклится

2 Дан код:

```
class Book {
private String book;
public void setBook (String b) {book = b;}
}
public class Quest7 {
public static void main (String [] args) {
Book book1 = new Book (); book1.setBook ("Java 7");
Book book2 = new Book (); book2.setBook ("Java 7");
if (book1.equals (book2)) {
System.out.println ("True");
```

```
} else {
System.out.println ("False");
}
}
}
```

Результатом компиляции и запуска кода будет?

a. True
b. ошибка компиляции
c. False
d. код скомпилируется, но при выполнении оператор if будет пропущен.

3 Укажите строки, компиляция которых не приведет к ошибке:

1) int var1 = 356f
2) **double var2 = 356f**
3) float var3 = 356f
4) char var4 = 356f
5) long var5 = 356f
6) byte var6 = 356f
7) Integer var7 = 356f
8) Character var8 = 356f
9**) Object var9 = 356f**

4 Дан код:

```
public class Quest4 {
public static void main (String [] args) {
double x=0, y=2, z;
z = y/x;
System.out.println ("z="+z);
}
}
```

Что выведется на консоль в результате компиляции и запуска программы?

1) Ошибка компиляции
2) **z=Infinity**
3) z=NaN
4) Ошибка времени выполнения java.lang.ArithmeticException

3.6 Перечень типовых практических заданий к зачету

(для оценки навыков и (или) опыта деятельности)

1 Дан код:

```
class Item {}
```

1) int [] mas1 = new int [24];
2) Integer mas2 [] = new Integer [24];
3) char [] mas3 = new Character [] {'a', 'b', 'c'};
4) Item [] mas4 = new Item {new Item (), new Item ()};
5) double [] mas5 = {5, 10, 15, 20};
6) int [] mas6 [] = new int [4] [5];
7) int mas7 [] [] = new int [4] [];

Компиляция каких строк приведет к ошибке?

2 Дан фрагмент кода

```
if (e1) if (e2) S1; else S2;
```

(e1, e2, S1, S2 — корректные java-выражения).

Какому другому фрагменту кода он эквивалентен?

1) if (e1) {if (e2) S1; else S2;}

2) if (e1) {if (e2) S1;} else S2;
3) if (e1) if (e2) S1; else; else S2;
4) if (e1) if (e2) S1; else S2; else;

3 Какие утверждения о классах-оболочках корректны?
1**) Классы оболочки Double, Long, Float размещаются в пакете java.util**
2) Объекты классов оболочек могут хранить те же значения, что и соответствующие им базовые типы
3) **Объекты классов-оболочек хранят изменяемые значения аналогично переменным базовых типов**
4) **Объекты классов-оболочек по умолчанию получают значение null**
5) **В классах оболочках определены методы преобразования к базовому типу**

4 Дан код:
class Item {}
1) int [] mas1 = new int [24];
2) Integer mas2 [] = new Integer [24];
3**) char [] mas3 = new Character [] {'a', 'b', 'c'};**
4) **Item [] mas4 = new Item {new Item (), new Item ()};**
5) double [] mas5 = {5, 10, 15, 20};
6) int [] mas6 [] = new int [4] [5];
7) int mas7 [] [] = new int [4] [];
Компиляция каких строк приведет к ошибке?

3.7 Перечень теоретических вопросов к экзамену

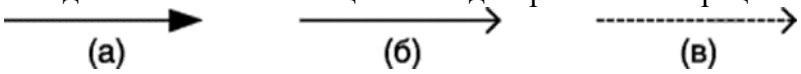
(для оценки знаний)

Раздел 3. «Введение в проектирование классов»
3.1. Типы отношений между классами.
3.2. UML принципы моделирования.
3.3. Виды диаграмм UML.
3.4. Элементы диаграммы Use Case.
3.5. Отношения между элементами диаграммы Use Case.
3.6. Назначение диаграмм Use Case.
3.7. Сценарий элемента Use Case.
3.8. Элементы диаграммы классов.
3.9. Стереотипы классов.
Раздел 4. «Использование классов и библиотек»
4.1. Размещение компонентов GUI Swing. Связь с архитектурой MVC. Обработка событий.
4.2. Параметризация типов и ее применение.
4.3. Потоки выполнения. Класс Tpread и интерфейс Runnable. Жизненный цикл потока.
4.4. Потоки-демоны. Методы synchronized.
4.5. Какие основные концепции работы с файлами Вы знаете?
4.6. Назовите несколько методов класса File.
4.7. Классы для работы с двоичными файлами.
4.8. Классы для работы с текстовыми файлами.
4.9. Виды кодировочных таблиц.
4.10. Работа с БД через JDBC.
4.11. Шаблон наблюдатель.

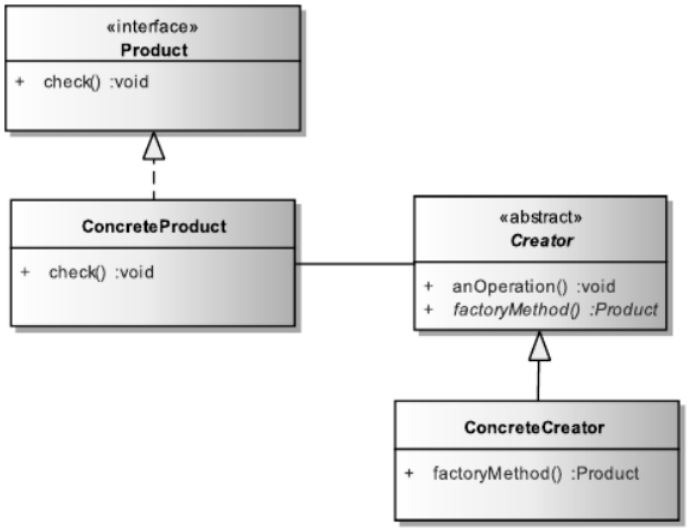
3.8 Перечень типовых простых практических заданий к экзамену

(для оценки умений)

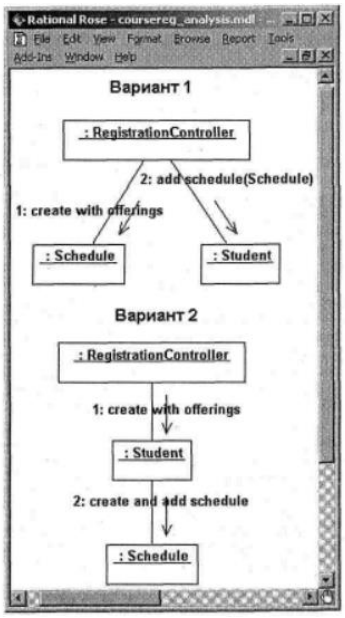
1 Подпишите типы сообщений на диаграмме кооперации



2 Какой шаблон представлен на рисунке?



3 Какой из вариантов создания графика соответствует шаблону Creator



4 Фрагмент кода использует шаблон _____?

```
public class Subject {
    Observers observers = new Observers();

    private Object field;

    public void setField(Object o) {
        field = o;
        observers.notifyObjectModified(this);
    }
}
```

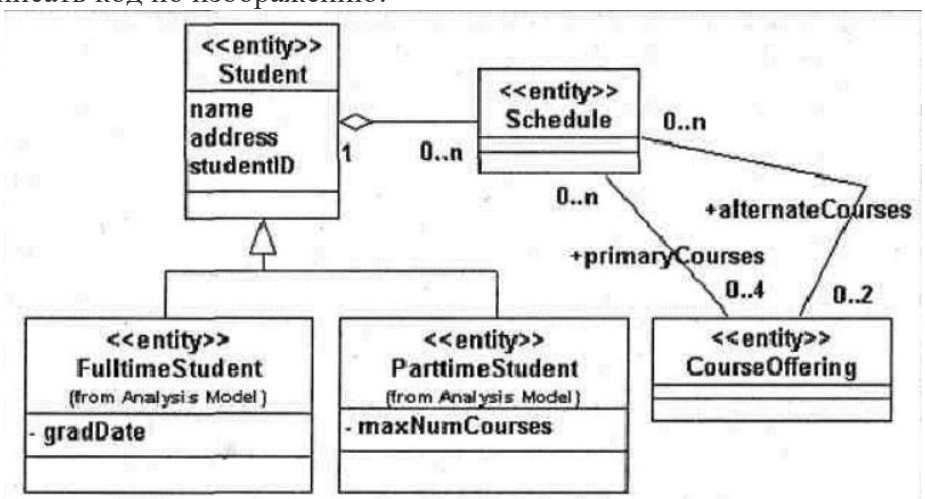
3.9 Перечень типовых практических заданий к экзамену

(для оценки навыков и (или) опыта деятельности)

1. Написать код класса Singleton для реализации фрагмента:

```
public class SingletonTest{
  public static void main(String[] args){
    Singleton ref = Singleton.getReference();
    System.out.println(ref.getValue());
    ref.setValue(ref.getValue() + 5);
    System.out.println(ref.getValue());
  }
```

2. Написать код по изображению:



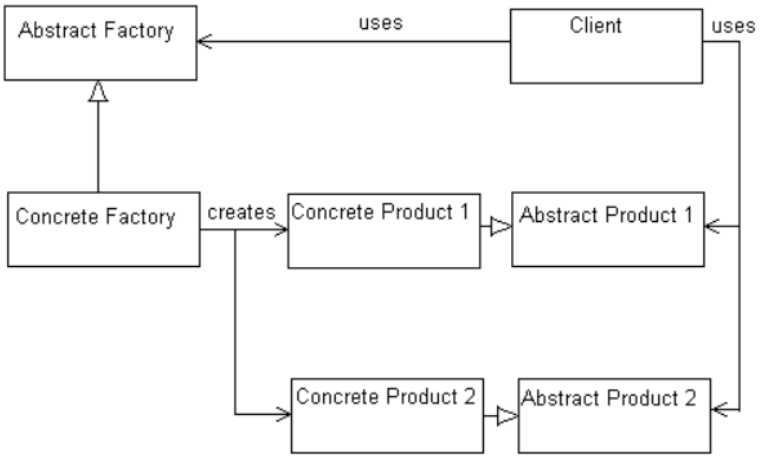
3. Объясните назначение класса EmptyObsorver.

```
public interface Observer {
    void objectCreated(Object obj);
    void objectModified(Object obj);
}

class EmptyObserver implements Observer {
    public void objectCreated(Object obj) { }
    public void objectModified(Object obj) { }
}

class Observers<T extends Observer> extends ArrayList<T> {
    public void notifyObjectCreated(Object obj) {
        for (Iterator<T> iter = (Iterator<T>) iterator(); iter.hasNext
            iter.next().objectCreated(obj);
    }
    public void notifyObjectModified(Object obj) {
        for (Iterator<T> iter = (Iterator<T>) iterator(); iter.hasNext
            iter.next().objectModified(obj);
    }
}
```

4. Рисунок использует шаблон_____?



4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности

В таблице приведены описания процедур проведения контрольно-оценочных мероприятий и процедур оценивания результатов обучения с помощью оценочных средств в соответствии с рабочей программой дисциплины.

Наименование оценочного средства	Описания процедуры проведения контрольно-оценочного мероприятия и процедуры оценивания результатов обучения
Собеседование	Собеседование, предусмотренное рабочей программой дисциплины, проводится на практическом занятии. Преподаватель на практическом занятии, предшествующем занятию проведения контроля, доводит до обучающихся тему, вопросы для подготовки к собеседованию. Результаты собеседования преподаватель доводит до обучающихся сразу после завершения собеседования
Тестирование (компьютерные технологии)	Тестирование проводится по результатам освоения тем или разделов дисциплины или по окончанию ее изучения во время практических занятий. Во время проведения тестирования пользоваться учебниками, справочниками, конспектами лекций, тетрадями для практических занятий не разрешено. Преподаватель на практическом занятии, предшествующем занятию проведения теста, доводит до обучающихся: темы, количество заданий в тесте, время выполнения. Результаты тестирования видны обучающемуся на компьютере сразу после прохождения теста
Лабораторная работа	Защита лабораторных работ проводится во время лабораторных занятий. Во время проведения защиты лабораторной работы пользоваться учебниками, справочниками, конспектами лекций, тетрадями не разрешено. Преподаватель на лабораторной работе, предшествующей занятию проведения защиты лабораторной работы, доводит до обучающихся: номер защищаемой лабораторной работы, время на защиту лабораторной работы. Преподаватель информирует обучающихся о результатах защиты лабораторной работы сразу после ее контрольно-оценочного мероприятия

Для организации и проведения промежуточной аттестации составляются 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.

Перечень теоретических вопросов и типовые практические задания разного уровня сложности для проведения промежуточной аттестации обучающиеся получают в начале семестра через электронную информационно-образовательную среду ИрГУПС (личный кабинет обучающегося).

Описание процедур проведения промежуточной аттестации в форме зачета и оценивания результатов обучения

При проведении промежуточной аттестации в форме зачета преподаватель может воспользоваться результатами текущего контроля успеваемости в течение семестра. С целью использования результатов текущего контроля успеваемости, преподаватель подсчитывает среднюю оценку уровня сформированности компетенций обучающегося (сумма оценок, полученных обучающимся, делится на число оценок).

Шкала и критерии оценивания уровня сформированности компетенций в результате изучения дисциплины при проведении промежуточной аттестации в форме зачета по результатам текущего контроля (без дополнительного аттестационного испытания)

Средняя оценка уровня сформированности компетенций по результатам текущего контроля	Шкала оценивания
Оценка не менее 3,0 и нет ни одной неудовлетворительной оценки по текущему контролю	«зачтено»

Оценка менее 3,0 или получена хотя бы одна неудовлетворительная оценка по текущему контролю	«не зачтено»

Если оценка уровня сформированности компетенций обучающегося не соответствует критериям получения зачета без дополнительного аттестационного испытания, то промежуточная аттестация проводится в форме собеседования по перечню теоретических вопросов и типовых практических задач или в форме компьютерного тестирования.

Промежуточная аттестация в форме зачета с проведением аттестационного испытания проходит на последнем занятии по дисциплине.

При проведении промежуточной аттестации в форме компьютерного тестирования вариант тестового задания формируется из фонда тестовых заданий по дисциплине случайным образом, но с условием: 50 % заданий должны быть заданиями открытого типа и 50 % заданий – закрытого типа.

Описание процедур проведения промежуточной аттестации в форме экзамена и оценивания результатов обучения

Промежуточная аттестация в форме экзамена проводится путем устного собеседования по билетам или в форме компьютерного тестирования.

При проведении промежуточной аттестации в форме собеседования билеты составляются таким образом, чтобы каждый из них включал в себя теоретические вопросы и практические задания.

Билет содержит: два теоретических вопроса для оценки знаний. Теоретические вопросы выбираются из перечня вопросов к экзамену; два практических задания: одно из них для оценки умений (выбирается из перечня типовых простых практических заданий к экзамену); другое практическое задание для оценки навыков и (или) опыта деятельности (выбираются из перечня типовых практических заданий к экзамену).

Распределение теоретических вопросов и практических заданий по экзаменационным билетам находится в закрытом для обучающихся доступе. Разработанный комплект билетов (25-30 билетов) не выставляется в электронную информационно-образовательную среду ИрГУПС, а хранится на кафедре-разработчике фондов оценочных средств.

На экзамене обучающийся берет билет, для подготовки ответа на экзаменационный билет обучающемуся отводится время в пределах 45 минут. В процессе ответа обучающегося на вопросы и задания билета, преподаватель может задавать дополнительные вопросы.

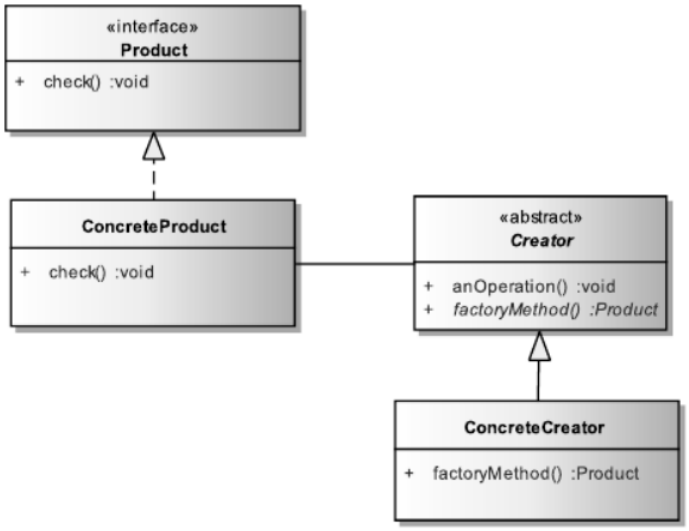
Каждый вопрос/задание билета оценивается по четырехбалльной системе, а далее вычисляется среднее арифметическое оценок, полученных за каждый вопрос/задание. Среднее арифметическое оценок округляется до целого по правилам округления

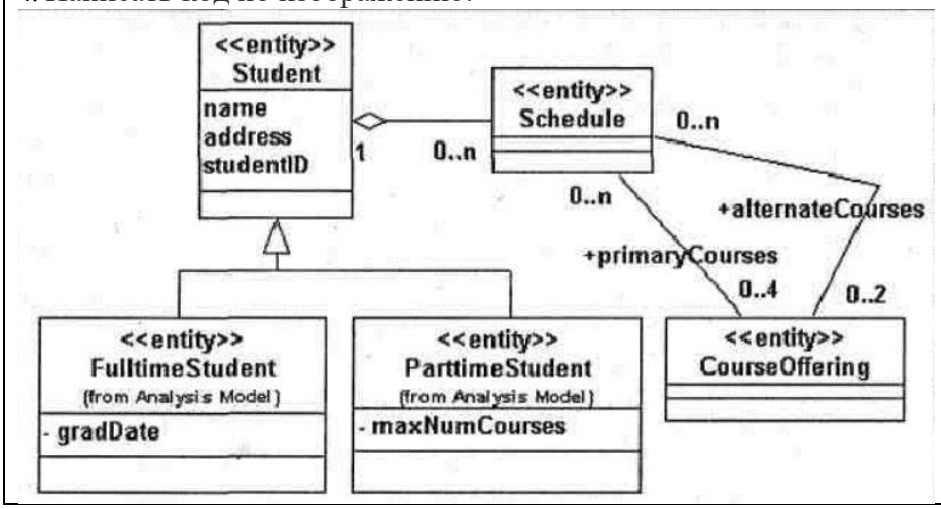
При проведении промежуточной аттестации в форме компьютерного тестирования вариант тестового задания формируется из фонда тестовых заданий по дисциплине случайным образом, но с условием: 50 % заданий должны быть заданиями открытого типа и 50 % заданий – закрытого типа.

Образец экзаменационного билета

ИРГУПС 20__-20__ учебный год	**Экзаменационный билет № 1** **по дисциплине «<u>Объектно-ориентированное программирование</u>»**	Утверждаю: Заведующий кафедрой «_____» ИрГУПС _____

1. Типы отношений между классами.
2. Классы для работы с двоичными файлами.
3. Какой шаблон представлен на рисунке?



…….

4. Написать код по изображению: