

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ  
БУРЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ ДОРЖИ БАНАЗАРОВА  
ИНСТИТУТ МАТЕМАТИКИ, ФИЗИКИ И КОМПЬЮТЕРНЫХ НАУК  
КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И ИНФОРМАТИКИ

Утверждена на заседании  
Ученого совета ИМФКН  
«\_\_»\_\_\_\_\_ 202\_\_ г.  
Протокол № \_\_

**Рабочая программа дисциплины**  
**Объектно-ориентированное проектирование**

Направление подготовки / специальность  
**09.04.02 Информационные системы и технологии**

Профиль  
**Проектирование, разработка и эксплуатация информационных систем**

Квалификация (степень) выпускника  
**Магистр**

Форма обучения  
**Очная**

Улан-Удэ  
2025

## Пояснительная записка

### Цели освоения дисциплины

Целью освоения дисциплины является изучение объектно-ориентированной методологии разработки и проектирования программных систем, а также в получении обучающимися навыков моделирования предметной области.

### Место дисциплины в структуре образовательной программы

Дисциплина Б1.О.01.03 «Объектно-ориентированное проектирование» относится к Блоку 1. Дисциплины (модули). Обязательная часть Б1.О.03. Информационные технологии учебного плана 09.04.02 Информационные системы и технологии. Предшествующими курсами, на которых непосредственно базируется дисциплина «Объектно-ориентированное проектирование» являются «Программирование», «Объектно-ориентированное программирование», «Проектирование информационных систем».

### В результате освоения дисциплины студент должен:

#### Планируемые результаты обучения по дисциплине и индикаторы достижения компетенций.

ОПК-2. Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач

ОПК-2.1. Понимает современные интеллектуальные технологии для решения профессиональных задач

ОПК-2.2. Обосновывает выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач

ОПК-8. Способен осуществлять эффективное управление разработкой программных средств и проектов

ОПК-8.1. Понимает методологические основы разработки программного обеспечения, методы управления проектами разработки программного обеспечения, нормативно-технические документы (стандарты и регламенты) по разработке программных средств и проектов

#### Знать:

методы построения объектно-ориентированных моделей информационно-телекоммуникационных систем в целях моделирования требований

#### Уметь:

проводить разработку и исследование теоретических и экспериментальных моделей объектов профессиональной деятельности

#### Владеть:

навыками по оценке качества объектно-ориентированного программного кода в современных операционных системах

#### Планируемые результаты освоения образовательной программы:

**Объем дисциплины в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем и на самостоятельную работу обучающихся**

Общая трудоемкость дисциплины составляет 4 зачетные единицы, 144 часа.

№ Название разделов дисциплины	Лекция	Лабораторная работа	Самостоятельная работа
Семестр 1	26	26	92
1 Современные методологии анализа и	26	26	92

## Тематическое планирование курса

### Темы

## Современные методологии анализа и разработки программных систем

Семестр 1

### Методологии анализа и разработки и программных систем

Лекция. 2(0) ч. Эволюция структуры построения программных систем

Лабораторная работа. 2(0) ч. Изучение основных принципов объектно-ориентированного анализа

Самостоятельная работа. 8(0) ч. Основные принципы объектно-ориентированного анализа

Лекция. 4(0) ч. Структура современной программной системы и ее описание на UML

Лабораторная работа. 4(0) ч. Изучение архитектуры построения информационных систем с точки зрения вариантов использования.

Самостоятельная работа. 22(0) ч. Язык графического описания UML

### Основы разработки объектно-ориентированных ИС

Лекция. 6(0) ч. Базовые концепции моделирования

Лекция. 6(0) ч. Особенности объектно-ориентированного программирования и механизмы объектно-ориентированных языков

Лекция. 4(0) ч. Базовые принципы проектирования объектно-ориентированных программ

Лабораторная работа. 10(0) ч. Изучение архитектуры построения информационных систем с точки зрения проектирования

Самостоятельная работа. 32(0) ч. Особенности проектирования структуры объектно-ориентированных программных систем

### Специальные шаблоны проектирования

Лекция. 4(0) ч. Специальные шаблоны проектирования

Лабораторная работа. 10(0) ч. Изучение специальных шаблонов проектирования информационных систем

Самостоятельная работа. 30(0) ч. Шаблоны проектирования информационных систем

## БРС

Семестр	Контрольные точки	Баллы
1	<b>Текущий контроль</b> в разделе «Современные методологии анализа и разработки программных систем»	
	Контрольная работа	30
1	<b>Текущий контроль</b> в разделе «Современные методологии анализа и разработки программных систем»	
	Контрольная работа	30

Семестр	Контрольные точки	Баллы
1	<b>Экзамен</b>	
	Устный ответ на вопрос экзаменационного билета	15
	Решение комплектов задач	25
Итого за семестр 1:		100

## Учебно-методическое и информационное обеспечение учебного процесса

### Образовательные технологии (в том числе на занятиях, проводимых в интерактивных формах).

Теоретическая часть курса, общие вопросы объектно-ориентированного проектирования излагаются в лекционном курсе. Отдельные вопросы могут выноситься на самостоятельное изучение. Для приобретения навыков работы на ПК предназначены лабораторные занятия. При изучении дисциплины используются интерактивные формы занятий (лекция-дискуссия, защита рефератов) в объеме 10 часов.

- [Методические рекомендации.doc](#)  
Режим доступа: [https://my.bsu.ru/content/file/2/21/213/818322\\_metodicheskie-rekomendacii.doc](https://my.bsu.ru/content/file/2/21/213/818322_metodicheskie-rekomendacii.doc)

### Учебно-методические материалы, в том числе методические указания для обучающихся по освоению дисциплины

Теоретическая часть курса, общие вопросы методики и технологий применения компьютерных средств излагаются преподавателем в лекционном курсе. Отдельные вопросы могут выноситься на самостоятельное изучение. Студент должен иметь в виду, что на лекциях преподаватель определяет такие вопросы и рекомендует необходимую для их изучения литературу, источники и др. ресурсы. Для успешного освоения курса необходимо внимательно фиксировать основные положения лекции, своевременно их усваивать, при необходимости самостоятельно прорабатывать, используя основную и дополнительную литературу.

Для приобретения навыков общения с ПК в процессе освоения инструментальных систем и отладки программ предназначены лабораторные занятия. Лабораторные занятия проводятся в специальных классах, оборудованных средствами вычислительной техники. На первом лабораторном занятии студенты получают инструктаж по технике безопасности при работе в классе и знакомятся с особенностями работы на конкретной вычислительной машине. Последующие лабораторные работы заключаются в освоении инструментальных систем и отладке программ решения типовых задач. Индивидуальные задания и методические указания к выполнению каждой последующей лабораторной работы студент получает, как правило, на предыдущем занятии. Подготовка к выполнению лабораторных работ осуществляется в часы самостоятельной работы. Студенты, не подготовившиеся к занятиям, к работе на компьютере не допускаются. По каждой выполненной лабораторной работе студент оформляет отчет по установленной форме.

Самостоятельные занятия под контролем преподавателя предназначены для самостоятельного изучения студентами тех разделов курса, по которым не предусмотрено

чтение лекций, либо проводятся лекции обзорного характера. По усмотрению преподавателя в часы индивидуальных занятий студентам может поручаться выполнение других заданий. Занятия проводятся с академической группой или с половиной группы в часы, установленные расписанием занятий. На занятиях студент должен иметь конспект лекций, учебную и справочную литературу, отдельную тетрадь для записей. Весь теоретический материал, изученный в процессе индивидуальных занятий, должен быть законспектирован.

### Оценочные средства

По данной дисциплине разработаны оценочные средства, критерии их оценивания, а также методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

- [Методические рекомендации.doc](#)
- [ФОС\\_Объектно-ориентир проектирование\\_Маги.docx](#)

### Список литературы

Перечень основной и дополнительной литературы, необходимой для освоения дисциплины.

#### Основная

1. [Программирование. Объектно-ориентированный подход](#): Учебник и практикум для вузов/Зыков С. В.. —Москва: Юрайт, 2021. —155 с.  
Режим доступа: <https://urait.ru/bcode/470281>
2. [Объектно-ориентированное программирование](#): Учебное пособие для вузов/Тузовский А. Ф.. —Москва: Юрайт, 2022. —206 с.  
Режим доступа: <https://urait.ru/bcode/490369>
3. Приемы объектно-ориентированного проектирования. Паттерны проектирования/Э. Гамма [и др.] ; [пер. с англ.: А. Слинкин]. —СПб.: Питер, 2012 [т.е. 2011]. —366 с.
4. [Проектирование информационных систем](#): Учебник и практикум для вузов/Трекул В. И., Коровкина Н. Л., Левочкина Г. А.. —Москва: Юрайт, 2022. —385 с.  
Режим доступа: <https://urait.ru/bcode/489918>
5. [Проектирование информационных систем. Методы и средства структурно-функционального проектирования. Практикум](#): учебное пособие для спо/Гвоздева Т. В., Баллод Б. А.; Гвоздева Т. В.. —Санкт-Петербург: Лань, 2023. —148 с.  
Режим доступа: <https://e.lanbook.com/book/314777>

#### Дополнительная

1. [Корпоративные информационные системы: требования при проектировании](#): Учебное пособие для вузов/Астапчук В. А., Терещенко П. В.. —Москва: Юрайт, 2022. —113 с.  
Режим доступа: <https://urait.ru/bcode/492141>
2. [Проектирование информационных систем](#): Учебник и практикум для вузов/под общ. ред. Чистова Д.В.. —Москва: Юрайт, 2022. —258 с.  
Режим доступа: <https://urait.ru/bcode/489307>

3. Объектно-ориентированное программирование: учебник/Барков И. А.. —Санкт-Петербург: Лань, 2022. —700 с.  
Режим доступа: <https://e.lanbook.com/book/206699>

**Перечень ресурсов информационно-коммуникационной сети «Интернет», необходимых для освоения дисциплины**

Sun Microsystems, Inc. The Java Tutorial - Режим доступа:

Sun Microsystems, Inc. JDK 6 Documentation - Режим доступа:

Электронный конспект лекций.

Тесты для компьютерного тестирования.

Федеральный портал. Российское образование. <http://www.edu.ru/>

Российский образовательный портал. <http://www.school.edu.ru/default.asp>

Естественный научно-образовательный портал. <http://www.en.edu.ru/>

Информационно-коммуникационные технологии в образовании. <http://www.ict.edu.ru/>

Наборы презентаций для лекционных занятий.

**Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)**

Портал электронного обучения БГУ [e.bsu.ru](http://e.bsu.ru)

Система дифференцированного интернет-обучения Nacadem, [Moodle.bsu.ru](http://Moodle.bsu.ru)

Личный кабинет преподавателя или студента БГУ <http://my.bsu.ru/>

Федеральное интернет-тестирование: проекты «Интернет-тренажеры в сфере профессионального образования» и «Федеральный интернет-экзамен в сфере профессионального образования»

База данных «Университет»

Электронные библиотечные системы: Руконт, издательство «Лань», Консультант студента

Программное обеспечение:

Office Standard 2007 Г/К 0302100006211000012-0001147-01 от 11.04.2011, бессрочная лицензия

Windows 7 Корпоративная Договор 46388/ИПК3863 от 03.04.2014 (Права на программы для ЭВМ Dream Spark Premium Electronic Software Delivery (3 years) )Обновляется один раз в 3 года

Java SE Development Kit 8бесплатные, ежегодно обновляемые, режим доступа:

<http://www.oracle.com/technetwork/java/javase/terms/license/index.html>

Net Beans IDE 8бесплатные, ежегодно обновляемые, режим доступа:

<https://netbeans.org/about/legal/product-licences.html>

Тестовый доступ: American Institute of Physics, Znaniun.com, Casc, Редакция журналов BMJ Group, БиблиоРоссика, электронная коллекция книг и журналов Informa Healthcare, Polpred, Science Translational Medicine, коллекция журналов BMG Group

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

Аудитория 0419

учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещение для самостоятельной работы

Оборудование:

персональный компьютер - 13 шт.

интерактивная система - 1 шт.

принтер - 1 шт.

переносной мультимедийный проектор - 1 шт.

переносной широкоформатный экран - 1 шт.

ноутбук переносной - 1 шт.  
системный блок - 1 шт.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГБОУ ВО «БУРЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ ДОРЖИ БАНЗАРОВА»

**Фонд оценочных средств по дисциплине**  
**Объектно-ориентированное проектирование**

Направление подготовки/ специальность  
**09.04.02** – Информационные системы и технологии

Профиль подготовки /специализация  
Проектирование, разработка и эксплуатация информационных систем

Квалификация (степень) выпускника  
магистр

Форма обучения  
очная



### Паспорт фонда оценочных средств

ОПК-2.1. Знает современные интеллектуальные технологии для решения профессиональных задач  
 ОПК-2.2 Умеет обосновывать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач  
 ОПК-8.1. Понимает методологические основы разработки программного обеспечения, методы управления проектами разработки программного обеспечения, нормативно-технические документы (стандарты и регламенты) по разработке программных средств и проектов

№	Контролируемые разделы, темы, модули	Формируемые компетенции	Этапы формирован ия	Оценочные средства	
				Вид	Количество
1.	Методологии анализа и разработки и программных систем	ОПК-2.1 ОПК-8.1	1 семестр	Контрольная работа	1
2.	Основы разработки объектно- ориентированных ИС. Специальные шаблоны проектирования	ОПК-2.2 ОПК-8.1	1 семестр	Контрольная работа	1
3.	Экзамен	ОПК-2.1 ОПК-2.2 ОПК-8.1	1 семестр	Устный ответ на вопрос экзаменационного билета Решение комплектов задач	1  1
Всего:				3	4

### Описание показателей и критериев оценивания уровня приобретенных компетенций на различных этапах их формирования

Результаты обучения	Показатели оценивания	Критерии сформированности	Шкала оценивания
<b>Знает:</b> методы построения объектно-ориентированных моделей информационно-телекоммуникационных систем в целях моделирования требований <b>Умеет:</b> проводить разработку и исследование теоретических и экспериментальных моделей объектов профессиональной деятельности <b>Владеет:</b> навыками по оценке качества объектно-ориентированного программного кода	<b>Пороговый уровень</b> (как обязательный для всех студентов)	<b>Знает:</b> основные принципы объектно-ориентированного анализа; <b>Умеет:</b> проводить структурирование современной программной системы; <b>Владеет:</b> знаниями об архитектуре построения информационных систем с точки зрения вариантов использования.	60-69 баллов
	<b>Базовый уровень</b>	<i>помимо удовлетворения критериям порогового уровня:</i> <b>Знает:</b> основные правила построения диаграмм классов и объектов; <b>Умеет:</b> проводить описание программной системы на UML; <b>Владеет:</b> методологией проектирования программных систем	70-84 баллов
	<b>Высокий уровень</b>	<i>помимо удовлетворения критериям порогового и базового уровня:</i> <b>Знает:</b> основные специальные шаблоны проектирования; <b>Умеет:</b> на основе анализа разрабатываемой задачи (программы) выбирать наиболее рациональные и экономичные структуры данных,	85-100 баллов

в современных операционных системах		обеспечивающие эффективную реализацию задачи (программы); <b>Владеет:</b> методологией проектирования программных систем со сложной организацией данных	
-------------------------------------	--	--	--

**Балльно-рейтинговая система**  
по дисциплине «Объектно-ориентированное проектирование»

Общая максимальная сумма баллов, которую студент может набрать по дисциплине в течение семестра – 100 баллов: 60 баллов текущий контроль и рубежный контроль + 40 баллов зачет/экзамен (итоговый контроль);

– общая максимальная сумма баллов, которую студент может набрать в течение семестра за выполнение всех видов работ во время аудиторных и внеаудиторных занятий, активность и посещаемость, должна быть равна 60 баллам;

– минимальная сумма баллов, при которой студент допускается к зачету/экзамену (итоговому контролю), равна 36 баллам (60% от 60 баллов);

– минимальная сумма баллов, при которой студент получает положительную итоговую оценку по дисциплине равна 60 баллам (60% от 100 баллов).

**Связь между четырехбалльной и столбальной системами  
оценки качества обучения студентов**

Оценка	Буквенный эквивалент оценки	Рейтинговые баллы
Отлично	A+	95-100
	A	90-94
	A-	85-89
Хорошо	B+	80-84
	B	75-79
	B-	70-74
Удовлетворительно	C+	67-69
	C	64-66
	C-	60-63
Неудовлетворительно	D	40-59
–	F	<40
Зачтено	S	60-100
Не зачтено	U	<60

**Оценочные средства и критерии их оценки**

**Примерные вопросы к экзамену:**

1. Назначение и структура языка UML.
2. Правила построения диаграмм классов и объектов.
3. Принцип единственности ответственности и шаблон проектирования Expert.
4. Шаблоны проектирования HighCohesion и LowCoupling
5. Шаблон проектирования Creator
6. Классический полиморфизм на основе наследования и его применение в базовых принципах проектирования.
7. Обоснованность применения наследования или композиции классов.
8. Отрицательное правило наследования
9. Понятие и назначение интерфейса. Отличие реализации интерфейса от наследования.
10. Выбор предпочтения между наследованием и реализацией интерфейса.
11. Состав и назначение SOLID-принципов

### **Критерии оценки теоретической части:**

- оценка «отлично» (14-15 баллов) *выставляется студенту, если он*
  - Четко знает основные принципы объектно-ориентированного анализа, умеет проводить структурирование современной программной системы, овладел умением методологией проектирования программных систем со сложной организацией данных.
  - Дает четкий и правильный ответ, выявляющий понимание учебного материала и характеризующий прочные знания, излагает материал в логической последовательности с использованием принятой терминологии;
  - Ошибок не делает, но допускает оговорки по невнимательности при работе с программными продуктами, которые легко исправляет по требованию преподавателя;
  - Ответ логичен, последователен, технически грамотен.
- оценка «хорошо» (11-13 баллов) *выставляется студенту, если он*
  - Овладел программным материалом, ориентируется в основных принципах объектно-ориентированного анализа, умеет проводить структурирование современной программной системы с небольшим затруднением;
  - Дает правильный ответ в определенной логической последовательности;
- оценка «удовлетворительно» (9-10 баллов) *выставляется студенту, если он*
  - Основной программный материал знает нетвердо, но большинство изученных понятий и обозначений усвоил;
  - Ответ дает неполный, построенный несвязно, но выявивший общее понимание вопросов;
- оценка «неудовлетворительно» (0-8 баллов) *выставляется студенту, если он*
  - Обнаруживает незнание или непонимание большей или наиболее важной части учебного материала;
  - Ответы строит несвязно, допускает существенные ошибки, которые не может исправить даже с помощью преподавателя.

### **Примерные задачи к экзамену:**

1. Терапевт Therapist ведет прием посетителей Person, в ходе которого выписывает лекарства Medicine; посетители принимают лекарства.
  - а. Постройте логическую модель, включающую классы Therapist, Medicine и Person, и отношения между ними.
  - б. Используя кооперации, покажите, что на приеме терапевт выполняет обязанности врача Doctor, посетитель является пациентом Patient, лекарства выписываются в виде рецептов Subscription.
2. Автомобиль Car состоит из двигателя класса Engine, пары передних front и задних rear колес класса Wheel.
  - а. Добавьте привод drivetrain так, чтобы автомобиль был переднеприводным.
  - б. Расширьте модель так, чтобы наряду с переднеприводными автомобилями, она описывала полноприводные автомобили как частный случай переднеприводных. Добавьте необходимые элементы, используйте двигатель DoubleEngine с двумя приводами типа drivetrain.
3. Пассажир Person заходит в лифт и нажимает кнопку pressButton лифта Lift с указанием целочисленного номера этажа floor. Лифт закрывает двери и начинает движение синхронным вызовом операции startMoving. После этого сообщает пассажиру номера проезжаемых лифтом этажей сообщением floorReached с указанием номера этажа. Затем лифт вызывает операцию stopMoving и останавливается. Пассажир нажимает кнопку pressDoors лифта для открытия дверей.
  - а. Как можно уточнить модель взаимодействия, если известно, что лифт обслуживает с первого по пятый этажи?
  - б. Уточните взаимодействие пассажира с лифтом. Укажите, что до нажатия кнопки этажа, пассажир обязан закрыть двери кнопкой pressDoors.
4. Пассажир Person заходит в лифт и нажимает кнопку pressButton лифта Lift с указанием целочисленного номера этажа floor. Лифт закрывает двери и начинает

движение синхронным вызовом операции `startMoving`. После этого сообщает пассажиру номера проезжаемых лифтом этажей сообщением `floorReached` с указанием номера этажа. Затем лифт вызывает операцию `stopMoving` и останавливается. Пассажир нажимает кнопку `pressDoors` лифта для открытия дверей.

а. Используя фрагменты, покажите, что пассажир не может нажать кнопку открытия и закрытия дверей в процессе движения лифта.

б. Укажите, что лифт проезжает один этаж за три секунды.

5. Интерфейс `Stack` определяет операции помещения в стек `push` с параметром `obj` типа `Element`, операцию получения элемента из стека `pop` с возвращаемым значением типа `Element`. При решении задачи используйте диаграммы классов.

а. Добавьте в интерфейс `Stack` операции очистки стека `reset`, которая не имеет параметров, статическую операцию создания нового стека `createNew` с возвращаемым значением типа `Stack`.

б. Покажите, что интерфейс `Stack` зависит от типа данных `Element`.

в. Добавьте класс `ListStack`, который реализует интерфейс `Stack`. Покажите реализуемые классом операции интерфейса.

6. Интерфейс `Stack` определяет операции помещения в стек `push` с параметром `obj` типа `Element`, операцию получения элемента из стека `pop` с возвращаемым значением типа `Element`. При решении задачи используйте диаграммы классов.

а. Добавьте в класс `ListStack` частное структурное свойство `arr` типа `Element` с кратностью больше нуля, значения которого упорядочены и могут повторяться.

б. Добавьте частный целочисленный атрибут `increment` только для чтения и защищенную операцию изменения размера `resize` с целочисленным параметром `newSize`.

в. Покажите на диаграмме экземпляров экземпляр `stack` класса `ListStack`, свойство `arr` которого содержит элемент `first` типа `Element` первым и `second` того же типа вторым. Укажите, что атрибут `increment` экземпляра `stack` равен 10.

7. В пространстве имен `Time` расположены перечисления `Month`, `DayOfWeek`, а также классы `Date` и `Period`. При решении задачи используйте диаграммы классов.

а. Укажите, что перечисление `Month` может принимать значения: `Jan`, `Feb`, `Mar`, `Apr`, `May`, `Jun`, `Jul`, `Aug`, `Sep`, `Oct`, `Nov`, `Dec`.

б. Укажите, что перечисление `DayOfWeek` может принимать значения: `Mon`, `Tue`, `Wed`, `Thu`, `Fri`, `Sat`, `Sun`.

в. Добавьте классу `Period` общедоступную статическую операцию `between`. У операции два аргумента: `from` и `to`. Оба аргумента имеют тип `Date`. Операция возвращает значение типа `Period`.

8. В пространстве имен `Time` расположены перечисления `Month`, `DayOfWeek`, а также классы `Date` и `Period`. При решении задачи используйте диаграммы классов.

а. Добавьте классу `Date` частные атрибуты `year`, `month`, `dayOfMonth` типа `Integer`, а также общедоступные операции:

- получения года `getYear` типа `Integer`;
- получения месяца `getMonth` типа `Month`;
- получения дня `getDayOfMonth` типа `Integer`;
- получения дня недели `getDayOfWeek` типа `DayOfWeek`.

б. Добавьте классу `Date` общедоступную статическую операцию `now` () типа `Date`.

в. Добавьте классу `Period` общедоступную статическую операцию `between`. У операции два аргумента: `from` и `to`. Оба аргумента имеют тип `Date`. Операция возвращает значение типа `Period`.

9. Абстрактный класс `Account` имеет два дочерних класса: счет физического лица `PersonalAccount` и юридического `CompanyAccount`. При решении задачи используйте диаграммы классов.

а. Добавьте класс `Person` с общедоступным атрибутом `FullName` строкового типа и свяжите его с классом `PersonalAccount` ассоциацией `Owns` с полюсом `owner` у `Person` и навигируемым полюсом `account` у `PersonalAccount`.

б. Аналогично для счета юридического лица добавьте владельца `Company`, свяжите анонимной ассоциацией с `CompanyAccount` и укажите подходящие названия полюсов.

в. Добавьте класс адреса `Address` с атрибутами строкового типа `street`, `city` и целочисленным положительным `building`. Укажите с помощью новых анонимных ассоциаций, что `Person` может иметь адрес регистрации `registeredAt`, фактический адрес `actual`, в то время как

компания связана с одним юридическим адресом legalAddress и может иметь почтовый адрес postAddress.

#### **Критерии оценки практической части:**

- оценка «отлично» (23-25 баллов)* выставляется студенту, если
- задача решена полностью;
  - в графическом изображении (UML), в теоретических выкладках решения нет пробелов и ошибок;
- оценка «хорошо» (18-22 баллов)* выставляется студенту, если
- задача решена полностью, но обоснования шагов решения недостаточны (если умение обосновывать рассуждения не являлось специальным объектом проверки);
  - допущена одна ошибка или два-три недочета в UML.
- оценка «удовлетворительно» (15-17 баллов)* выставляется студенту, если
- допущены более одной ошибки или двух-трех недочетов в UML, но студент владеет обязательными умениями по проверяемой теме;
  - студент не совсем понимает суть задачи и предлагает алгоритм, раскрывающий часть хода ее решения, но при помощи преподавателя исправляет указанные недоработки
- оценка «неудовлетворительно» (0-14 баллов)* выставляется студенту, если
- допущены существенные ошибки, показавшие, что студент не владеет обязательными знаниями по данной теме в полной мере.

### **Примерные темы контрольных работ**

#### **Контрольная работа №1**

1 вариант

Построить базовую диаграмму прецедентов для маршрутного такси

2 вариант

Многоугольник знает 1-ю и последнюю точки. Каждая точка имеет две координаты и знает следующую точку. Построить диаграмму классов и диаграмму объектов квадрата.

3 вариант

Тетрадь состоит из обложки, 2 скрепок и 4 листов. Тетради бывают в клетку и линейку. Построить диаграммы классов и объектов тетради в линейку.

4 вариант

Футбольная команда состоит из 11 игроков. Игроки могут быть нападающими, полузащитниками и защитниками. Один из игроков является капитаном. Построить диаграмму классов.

5 вариант

Нарисовать диаграмму последовательностей разъединения абонентов в предположении, что первым трубку кладет абонент A1.

#### **Критерии оценки:**

- оценка «отлично» (28-30 баллов)* выставляется студенту, если
- работа выполнена полностью;
  - в диаграмме класса нет ошибок;
- оценка «хорошо» (23-27 баллов)* выставляется студенту, если
- работа выполнена полностью, но обоснования шагов решения недостаточны (если умение обосновывать рассуждения не являлось специальным объектом проверки);
  - допущена одна ошибка или два-три недочета в диаграмме класса.
- оценка «удовлетворительно» (15-22 балла)* выставляется студенту, если
- допущены более одной ошибки или двух-трех недочетов в диаграмме класса, но студент владеет обязательными умениями по проверяемой теме.
- оценка «неудовлетворительно» (0-14 баллов)* выставляется студенту, если
- допущены существенные ошибки, показавшие, что студент не владеет обязательными знаниями по данной теме в полной мере.

#### **Контрольная работа №2**

#### 1 вариант

1. Массив Table состоит из нескольких элементов Element.

а. Укажите, что элементы упорядочены и могут повторяться.

б. Отрадите в модели, что имея экземпляр массива, можно перейти к его элементам,

но от экземпляра элемента нельзя перейти к массиву.

в. Пусть массив проиндексирован таким образом, что индексом элемента также является

элемент. Используя квалификаторы, отразите данное свойство в модели.

#### 2 вариант

Преподаватель Teacher ведет teaches несколько курсов CourseOffering.

а. Используя агрегацию, покажите, что курс состоит из одной лекции Lecture и нескольких семинаров Practice.

б. Укажите, что преподаватель ведет семинары как ассистент assistant и читает лекции как лектор lecturer.

в. (\*) Измените свойство teaches так, чтобы оно всегда указывало только на курсы, по которым преподаватель читает лекции или ведет семинары.

#### 3 вариант

Каждый экземпляр абстрактного класса контроллер Controller связан по ассоциации Sensor с несколькими датчиками поезда TrainSensor. В ассоциации контроллер играет роль управляющего controller. Датчик поезда участвует в ассоциации как датчик sensor с частной видимостью.

а. Используя квалификаторы, укажите, что каждому значению индекса index типа String соответствует не более одного датчика в ассоциации Sensor.

б. Измените класс контроллера, укажите, что класс принимает сигналы приближения поезда TrainSpotted и отдаления поезда TrainLeft, имеет общедоступную операцию выполнения команд execute с параметром команда cmd типа данных Command и возвращает значение типа данных Result.

в. Определите класс цифрового контроллера DigitalController, уточняющий класс контроллера. В классе цифрового контроллера определена операция executeDigital, которая переопределяет операцию выполнения команд контроллера и возвращает цифровой результат DigitalResult.

г. Используя экземпляры классов, приведите пример контроллера с двумя датчиками.

#### 4 вариант

Игрок Player заключает контракт Contract с командой Team. Команда может заключить до двадцати контрактов с разными игроками. Контракт заключается на определенный срок period с компенсацией salary (класс ассоциации). В контракте игрок указан работником worker, команда – нанимателем employer.

а. Добавьте в контракт пункты Item, каждый из которых содержит текст statement. Укажите, что менеджер Manager управляет manages контрактами.

б. Укажите, что игрок может быть нападающим Forward, защитником Guard или центровым Center. Игрок не может иметь несколько специализаций.

в. Игроки реагируют на команды тренера Coach. Во время игры, тренер может отправлять им указания: нападать attack, перейти к обороне guard, играть совместно join с другим игроком peer. Работа тренера состоит во взаимодействии и обучении Train команды.

г. Измените модель таким образом, чтобы игроки могли изменять специализацию: нападающий, защитник или центровой.

**Критерии оценки:**

*оценка «отлично» (28-30 баллов)* выставляется студенту, если

- работа выполнена полностью;
- в описании класса нет ошибок;

*оценка «хорошо» (23-27 баллов)* выставляется студенту, если

• работа выполнена полностью, но обоснования шагов решения недостаточны (если умение обосновывать рассуждения не являлось специальным объектом проверки);

- допущена одна ошибка или два-три недочета в описании класса.

*оценка «удовлетворительно» (15-22 балла)* выставляется студенту, если

• допущены более одной ошибки или двух-трех недочетов в описании класса, но студент владеет обязательными умениями по проверяемой теме.

*оценка «неудовлетворительно» (0-14 баллов)* выставляется студенту, если

• допущены существенные ошибки, показавшие, что студент не владеет обязательными знаниями по данной теме в полной мере.