

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Национальный исследовательский ядерный университет «МИФИ»  
**Обнинский институт атомной энергетики** –  
филиал федерального государственного автономного образовательного учреждения высшего образования  
«Национальный исследовательский ядерный университет «МИФИ»  
**(ИАТЭ НИЯУ МИФИ)**

Одобрено на заседании  
Ученого совета ИАТЭ НИЯУ МИФИ  
Протокол от 24.04.2023 № 23.4

## **ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

по учебной дисциплине

---

Программирование и моделирование

*название дисциплины*

для студентов направления подготовки

03.03.02 Физика

---

*код и название [специальности/направления подготовки]*

образовательная программа

---

Ядерно-физические технологии в медицине

Форма обучения: очная

**г. Обнинск 2023 г.**

## Область применения

Фонд оценочных средств (ФОС) – является неотъемлемой частью учебно-методического комплекса учебной дисциплины «Программирование и моделирование» и предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу данной дисциплины.

## Цели и задачи фонда оценочных средств

Целью Фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям федерального государственного образовательного стандарта.

Для достижения поставленной цели Фондом оценочных средств по дисциплине «Программирование и моделирование» решаются следующие задачи:

- контроль и управление процессом приобретения обучающимися знаний, умений и навыков предусмотренных в рамках данного курса;
- контроль и оценка степени освоения компетенций предусмотренных в рамках данного курса;
- обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения в образовательный процесс в рамках данного курса.

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

### *Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы*

В результате освоения ООП бакалавриата обучающийся должен овладеть следующими результатами обучения по дисциплине:

Коды компетенций	Результаты освоения ООП. Содержание компетенций	Перечень планируемых результатов обучения по дисциплине
УКЕ-1	Способен использовать знания естественнонаучных дисциплин, применять методы математического анализа и моделирования, теоретического и экспериментального исследования в поставленных задачах.	З-УКЕ-1 знать: основные законы естественнонаучных дисциплин, методы математического анализа и моделирования, теоретического и экспериментального исследования У-УКЕ-1 уметь: использовать математические методы в технических приложениях, рассчитывать основные числовые характеристики случайных величин, решать основные задачи математической статистики; решать типовые расчетные задачи В-УКЕ-1 владеть: методами математического анализа и моделирования; методами решения задач анализа и расчета характеристик физических систем, основными приемами обработки экспериментальных данных, методами работы с прикладными программными продуктами
УКЦ-1	Способен в цифровой среде использовать	З-УКЦ-1 Знать: современные информационные технологии и цифровые средства коммуникации, в том

Коды компетенций	Результаты освоения ООП. Содержание компетенций	Перечень планируемых результатов обучения по дисциплине
	различные цифровые средства, позволяющие во взаимодействии с другими людьми достигать поставленных целей.	числе отечественного производства, а также основные приемы и нормы социального взаимодействия и технологии межличностной и групповой коммуникации с использованием дистанционных технологий У-УКЦ-1 Уметь: выбирать современные информационные технологии и цифровые средства коммуникации, в том числе отечественного производства, а также устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе и применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды с использованием дистанционных технологий В-УКЦ-1 Владеть: навыками применения современных информационных технологий и цифровых средств коммуникации, в том числе отечественного производства, а также методами и приемами социального взаимодействия и работы в команде с использованием дистанционных технологий
УКЦ-2	Способен искать нужные источники информации и данные, воспринимать, анализировать, запоминать и передавать информацию с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач	З-УКЦ-2 Знать: методики сбора и обработки информации с использованием цифровых средств, а также актуальные российские и зарубежные источники информации в сфере профессиональной деятельности, принципы, методы и средства решения стандартных задач профессиональной деятельности с использованием цифровых средств и с учетом основных требований информационной безопасности У-УКЦ-2 Уметь: применять методики поиска, сбора и обработки информации; с использованием цифровых средств, осуществлять критический анализ и синтез информации, полученной из разных источников, и решать стандартные задачи профессиональной деятельности с использованием цифровых средств и с учетом основных требований информационной безопасности В-УКЦ-2 Владеть: методами поиска, сбора и обработки, критического анализа и синтеза информации с использованием цифровых средств для решения поставленных задач, навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций и библиографии по научно-исследовательской работе с использованием цифровых средств и с учетом требований информационной безопасности
ОПК-3	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач	З-ОПК-3 знать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности У-ОПК-3 уметь выбирать и использовать современные информационные технологии и программные средства для решения задач профессиональной деятельности

Коды компетенций	Результаты освоения ООП. Содержание компетенций	Перечень планируемых результатов обучения по дисциплине
	профессиональной деятельности	В-ОПК-3 владеть современными информационными технологиями и программными средствами при решении задач профессиональной деятельности, соблюдая требования информационной безопасности
ПК-7	Способен анализировать исходные данные проектирования, участвовать в разработке, подготовке и оформлении проектной документации с учетом норм радиационной и экологической безопасности	3-ПК-7 знать нормы радиационной и экологической безопасности, а также правила разработки, подготовки и оформления проектной документации с учетом норм радиационной и экологической безопасности У-ПК-7 уметь анализировать и критически оценивать любую поступающую информацию, выделять и систематизировать данные В-ПК-7 владеть навыками сбора, обработки, анализа и систематизации, а также оформления проектной документации с учетом норм радиационной и экологической безопасности

### ***Этапы формирования компетенций в процессе освоения ООП бакалавриата***

Компоненты компетенций, как правило, формируются при изучении нескольких дисциплин, а также в немалой степени в процессе прохождения практик, НИР и во время самостоятельной работы обучающегося. Выполнение и защита ВКР являются видом учебной деятельности, который завершает процесс формирования компетенций.

Место дисциплины и соответствующий этап формирования компетенций в целостном процессе подготовки по образовательной программе можно определить по матрице компетенций, которая приводится в Приложении.

Этапы формирования компетенции в процессе освоения дисциплины:

- **начальный** этап – на этом этапе формируются знаниевые и инструментальные основы компетенции, осваиваются основные категории, формируются базовые умения. Студент воспроизводит термины, факты, методы, понятия, принципы и правила; решает учебные задачи по образцу;

- **основной** этап – знания, умения, навыки, обеспечивающие формирование компетенции, значительно возрастают, но еще не достигают итоговых значений. На этом этапе студент осваивает аналитические действия с предметными знаниями по дисциплине, способен самостоятельно решать учебные задачи, внося коррективы в алгоритм действий, осуществляя коррекцию в ходе работы, переносит знания и умения на новые условия;

- **завершающий** этап – на этом этапе студент достигает итоговых показателей по заявленной компетенции, то есть осваивает весь необходимый объем знаний, овладевает всеми умениями и навыками в сфере заявленной компетенции. Он способен использовать эти знания, умения, навыки при решении задач повышенной сложности и в нестандартных условиях.

Этапы формирования компетенций в ходе освоения дисциплины отражаются в тематическом плане (см.п. 4 рабочей программы дисциплины).

### **Паспорт фонда оценочных средств по дисциплине «Технологии программирования»**

№ п/п	Контролируемые модули, разделы дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1.	Раздел 1. Спецификация требований и проектирование ПО	ОПК-8, ОПК-9, ПК-5	Лабораторная работа №1. Письменно.
2.	Раздел 2. Специализированные технологии создания ПО	ОПК-8, ОПК-9, ПК-5	Контрольная работа №1. Письменно. Лабораторная работа №2. Письменно.
3.	Раздел 3. Тестирование и документирование ПО	ОПК-8, ОПК-9, ПК-5	Лабораторная работа №3. Письменно.
4.	Раздел 4. Управление IT-проектами	ОПК-8, ОПК-9, ПК-5	Лабораторная работа №4. Письменно.

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

Конечными результатами освоения программы дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям, которые приведены в п.1.1. Формирование этих дескрипторов происходит в процессе изучения дисциплины по этапам в рамках различного вида учебных занятий и самостоятельной работы.

Выделяются три уровня сформированности компетенций на каждом этапе: пороговый, продвинутый и высокий.

Уровни	Содержательное описание уровня	Основные признаки выделения уровня	БРС, % освоения	ECTS/Пятибалльная шкала для оценки экзамена/зачета
<b>Высокий</b> <i>Все виды компетенций сформированы на высоком уровне в соответствии с целями и задачами дисциплины</i>	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Студент демонстрирует свободное обладание компетенциями, способен применить их в нестандартных ситуациях: показывает умение самостоятельно принимать решение, решать проблему/задачу теоретического или прикладного характера на основе изученных методов, приемов, технологий	90-100	A/ Отлично/ Зачтено
<b>Продвинутый</b> <i>Все виды компетенций сформированы на продвинутом уровне в соответствии с целями и задачами дисциплины</i>	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу, большей долей самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Студент может доказать владение компетенциями: демонстрирует способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения.	85-89	B/ Очень хорошо/ Зачтено
			75-84	C/ Хорошо/ Зачтено
<b>Пороговый</b> <i>Все виды компетенций сформированы на пороговом уровне</i>	Репродуктивная деятельность	Студент демонстрирует владение компетенциями в стандартных ситуациях: излагает в пределах задач курса теоретически и практически контролируемый материал.	65-74	D/Удовлетворительно / Зачтено
			60-64	E/Посредственно /Зачтено
<b>Ниже порогового</b>	Отсутствие признаков порогового уровня: компетенции не сформированы. Студент не в состоянии продемонстрировать обладание компетенциями в стандартных ситуациях.		0-59	Неудовлетворительно / Зачтено

Оценивание результатов обучения студентов по дисциплине осуществляется по регламенту текущего контроля и промежуточной аттестации.

Критерии оценивания компетенций на каждом этапе изучения дисциплины для каждого вида оценочного средства и приводятся в п. 4 ФОС. Итоговый уровень сформированности компетенции при изучении дисциплины определяется по таблице. При этом следует понимать, что граница между уровнями для конкретных результатов освоения образовательной программы может смещаться.

<b>Уровень сформированности компетенции</b>	<b>Текущий контроль</b>	<b>Промежуточная аттестация</b>
высокий	<b>высокий</b>	<b>высокий</b>
	<i>продвинутый</i>	<i>высокий</i>
	<i>высокий</i>	<i>продвинутый</i>
продвинутый	<i>пороговый</i>	<i>высокий</i>
	<i>высокий</i>	<i>пороговый</i>
	<b>продвинутый</b>	<b>продвинутый</b>
	<i>продвинутый</i>	<i>пороговый</i>
	<i>пороговый</i>	<i>продвинутый</i>
пороговый	<b>пороговый</b>	<b>пороговый</b>
ниже порогового	<b>пороговый</b>	<b>ниже порогового</b>
	<b>ниже порогового</b>	-

### **Методические материалы, определяющие процедуры оценивания знаний, умений, навыков или опыта деятельности, характеризующих этапы формирования компетенций.**

Рейтинговая оценка знаний является интегральным показателем качества теоретических и практических знаний и навыков студентов по дисциплине и складывается из оценок, полученных в ходе текущего контроля и промежуточной аттестации.

Текущий контроль в семестре проводится с целью обеспечения своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы студентов.

Промежуточная аттестация предназначена для объективного подтверждения и оценивания достигнутых результатов обучения после завершения изучения дисциплины.

Текущий контроль осуществляется два раза в семестр: контрольная точка № 1 (КТ № 1) и контрольная точка № 2 (КТ № 2).

Результаты текущего контроля и промежуточной аттестации подводятся по шкале балльно-рейтинговой системы.

**Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков. Экзаменационные билеты**

Федеральное государственное автономное образовательное учреждение высшего профессионального образования

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ  
«МИФИ»

ОБНИНСКИЙ ИНСТИТУТ АТОМНОЙ ЭНЕРГЕТИКИ

Кафедра «Компьютерные системы, сети и технологии»

09.03.01 «Информатика и вычислительная техника»

Дисциплина «Технологии программирования»

**Экзаменационный билет № 1**

**Вопрос №1**

Основные понятия ООП: абстракция, инкапсуляция, наследование, полиморфизм. Альтернативное определение ООП.

**Вопрос №2**

Java как язык ООП для машинно-независимых приложений.

**Вопрос № 3**

Требования к ПО. Их виды, методы сбора, текстовый анализ.

Составитель \_\_\_\_\_ Тельнов В.П.

Заведующий кафедрой \_\_\_\_\_ Старков С.О.

**Критерии оценки:**

100 - балл выставляется студенту, если дан 100% ответ на билет.

75 - балл выставляется студенту, если дан 75% ответ на билет.

60 - балл выставляется студенту, если дан 60% ответ на билет.

Неудовлетворительный балл выставляется студенту, если дан менее чем 60% ответ на билет.

## Прочие экзаменационные билеты

### Экзаменационный билет № 2

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Наследование. Множественное наследование.

Вопрос № 2

Обзор Java-технологий NetBeans IDE.

Вопрос № 3

Диаграммы компонентов. Нотация, семантика, примеры.

### Экзаменационный билет № 3

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Виртуальные функции. Абстрактные классы. Полиморфные контейнеры.

Вопрос № 2

Компонентные технологии разработки программного обеспечения.

Вопрос № 3

Диаграммы последовательности. Нотация, семантика, примеры.

### Экзаменационный билет № 4

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Конструкторы и деструкторы. Их назначение, виды, примеры.

Вопрос № 2

Обзор различий между Java и C++.

Вопрос № 3

Диаграммы развертывания. Нотация, семантика, примеры.

### Экзаменационный билет № 5

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Перегруженные операции. Их виды, примеры на языке C++.

Вопрос № 2

Требования к ПО. Их виды, методы сбора, текстовый анализ.

Вопрос № 3

Диаграммы прецедентов. Нотация, семантика, примеры.

### **Экзаменационный билет № 6**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Обработка нештатных ситуаций. Объекты-исключения. Примеры.

Вопрос № 2

Диаграммы последовательности. Нотация, семантика, примеры.

Вопрос № 3

Рефакторинг ПО. Общие принципы, использование паттернов.

### **Экзаменационный билет № 7**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Модели жизненного цикла программного обеспечения.

Вопрос № 2

Диаграммы компонентов. Нотация, семантика, примеры.

Вопрос № 3

Диаграммы классов. Нотация, семантика, примеры.

### **Экзаменационный билет № 8**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Проектирование программного обеспечения и UML.

Вопрос № 2

Диаграммы развертывания. Нотация, семантика, примеры.

Вопрос № 3

Технология JSP. Архитектура, директивы, теги, скриптлеты. Пример.

### **Экзаменационный билет № 9**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Диаграммы прецедентов. Нотация, семантика, примеры.

Вопрос № 2

Архитектурный паттерн MVC. Область применения, схема взаимодействия.

Вопрос № 3

Управление IT-проектами. Методология RAD.

### **Экзаменационный билет № 10**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Сценарии выполнения прецедентов. Пример.

Вопрос № 2

Технология JSP. Архитектура, директивы, теги, скриптлеты. Пример.

Вопрос № 3

Документирование ПО. Принципы и стандарты. Виды документов.

### **Экзаменационный билет № 11**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Диаграммы классов. Нотация, семантика, отношения.

Вопрос № 2

Объектно-реляционное отображение. Архитектура, пример маппинга.

Вопрос № 3

Управление IT-проектами. Экстремальное программирование.

### **Экзаменационный билет № 12**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Диаграммы классов. Атрибуты и операции.

Вопрос № 2

Маппинг коллекций с использованием аннотаций. Примеры отношений.

Вопрос № 3

C# как язык ООП для .NET Framework.

### **Экзаменационный билет № 13**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Отношения ассоциации на диаграммах классов. Примеры.

Вопрос № 2

Тестирование и качество ПО. Верификация, аттестация, инспектирование.

Вопрос № 3

C#: сборки, манифесты, домены, компоненты.

### **Экзаменационный билет № 14**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Отношения обобщения на диаграммах классов. Примеры.

Вопрос № 2

Виды, методы и цели тестирования ПО.

Вопрос № 3

Виртуальные функции. Абстрактные классы. Полиморфные контейнеры.

### **Экзаменационный билет № 15**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Отношения агрегации на диаграммах классов. Примеры.

Вопрос № 2

Тестирование «черного» и «белого» ящиков. Классы эквивалентности.

Вопрос № 3

Java как язык ООП для машинно-независимых приложений.

### **Экзаменационный билет № 16**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Отношения композиции на диаграммах классов. Примеры.

Вопрос № 2

Планирование, документирование, измерение процесса тестирования.

Вопрос № 3

Архитектурный паттерн MVC. Область применения, схема взаимодействия.

### **Экзаменационный билет № 17**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Обзор платформы .NET Framework.

Вопрос № 2

Рефакторинг ПО. Общие принципы, использование паттернов.

Вопрос № 3

Тестирование и качество ПО. Верификация, аттестация, инспектирование.

### **Экзаменационный билет № 18**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

C# как язык ООП для .NET Framework.

Вопрос № 2

Документирование ПО. Принципы и стандарты. Виды документов.

Вопрос № 3

Маппинг коллекций с использованием аннотаций. Примеры отношений.

### **Экзаменационный билет № 19**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

C#: сборки, манифесты, домены, компоненты.

Вопрос № 2

Управление IT-проектами. Методология RAD.

Вопрос № 3

Тестирование «черного» и «белого» ящиков. Классы эквивалентности.

### **Экзаменационный билет № 20**

по курсу «Технологии программирования» для специальности 230100

Вопрос № 1

Обзор различий между C# и C++.

Вопрос № 2

Управление IT-проектами. Экстремальное программирование.

Вопрос № 3

Объектно-реляционное отображение. Архитектура, пример маппинга.

## Перечень оценочных средств

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
1	2	3	4
1	Контрольная работа	Средство проверки умений применять полученные знания для решения задач определенного типа по теме или разделу	Комплект контрольных заданий по вариантам
2	Проект	Конечный продукт, получаемый в результате планирования и выполнения комплекса учебных и исследовательских заданий. Позволяет оценить умения обучающихся самостоятельно конструировать свои знания в процессе решения практических задач и проблем, ориентироваться в информационном пространстве и уровень сформированности аналитических, исследовательских навыков, навыков практического и творческого мышления. Может выполняться в индивидуальном порядке или группой обучающихся.	Темы групповых и/или индивидуальных проектов в форме заданий лабораторного практикума

Кафедра Компьютерных систем, сетей и технологий

**Комплект заданий для контрольной работы**

по дисциплине «Технология разработки программного обеспечения»

**Вариант 1**

1. Студент посещает несколько аудиторий. Аудитория может вмещать несколько студентов. Какими отношениями могут быть связаны данные классы? Дайте аргументированный ответ и приведите **варианты соответствующих диаграмм классов**.
2. **На языке Java опишите класс «процессор»**, содержащий сведения о его марке, тактовой частоте, объеме кэша и стоимости. Предусмотрите инициализацию с проверкой допустимости значений полей. В случае недопустимых значений полей выбрасываются исключения. Опишите методы для получения состояния объекта. **Опишите класс «материнская плата»**, включающий класс «процессор» и объем установленной оперативной памяти. Предусмотрите инициализацию с проверкой допустимости значений поля объема памяти. В случае недопустимых значений поля выбрасывается исключение. Опишите свойства для получения состояния объекта.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Процессор» и «Материнская плата», а также отношения между ними.

**Вариант 2**

**На языке Java опишите класс «самолет»**, содержащий следующие закрытые поля:

- название пункта назначения;
- шестизначный номер рейса;
- время отправления.

Предусмотрите методы для установки и получения состояния объекта.

**Опишите класс «аэропорт»**, содержащий закрытый массив самолетов. Обеспечьте следующие возможности:

- вывод информации о самолете по номеру рейса;
- вывод информации о самолетах, отправляющихся в течение часа после введенного с клавиатуры времени;
- вывод информации о самолетах, отправляющихся в заданный пункт назначения;
- сравнение времени отправления двух самолетов.

Вся выводимая информация должна быть отсортирована по времени отправления.

**Вариант 3**

**На языке Java опишите класс «Файл» на диске**. Файлы можно создавать, удалять, открывать, а также выполнять с ними операции записи и чтения. После создания файл может быть открыт приложением для записи или чтения. Открывать для записи файл может лишь одно приложение одновременно. Во время записи чтение из файла запрещено. Открыть для чтения файл могут несколько приложений одновременно. Запись в файл во время чтения запрещена. Жизненный цикл файла заканчивается при его удалении. Удалить можно лишь файлы не открытые для записи или чтения.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствует сам класс «Файл» и класс приложений, которые с ним работают, а также отношения между ними.

#### Вариант 4

**На языке Java опишите класс «комната»,** содержащий сведения о метраже, высоте потолков и количестве окон. Предусмотрите инициализацию с проверкой допустимости значений полей. В случае недопустимых значений полей выбрасываются исключения. Описать методы вычисления площади и объема комнаты и свойства для получения состояния объекта.

**Опишите класс «квартира»,** включающий класс «комната» и общую площадь квартиры. Предусмотрите инициализацию с проверкой допустимости значений поля «площадь квартиры». В случае недопустимых значений поля выбрасывается исключение. Опишите свойства для получения состояния объекта.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Комната» и «Квартира», а также отношения между ними.

#### Вариант 5

**На языке Java опишите контейнерный класс «автостоянка»** для хранения сведений об автомобилях. Для каждого объекта класса «Автомобиль» записываются его госномер, цвет, фамилия владельца и признак присутствия на стоянке. Предусмотрите инициализацию с проверкой допустимости значений полей. В случае недопустимых значений полей выбрасываются исключения. Обеспечьте возможность поиска автомобиля по разным критериям, вывода списка присутствующих и отсутствующих на стоянке автомобилей, доступа к имеющимся сведениям по номеру места на автостоянке.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Автостоянка» и «Автомобиль», а также отношения между ними.

#### Вариант 6

**На языке Java опишите контейнерный класс «Предметный указатель».** Предметный указатель содержит объекты класса «Слово», которые инкапсулируют сведения о номерах страниц в документе, на которых это слово встречается. Количество номеров страниц, относящихся к одному слову, от одного до десяти. Предусмотрите возможность формирования указателя для некоторого документа, вывода указателя, вывода номеров страниц для заданного слова, удаления элемента из указателя.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Предметный указатель» и «Слово», а также отношения между ними.

#### Вариант 7

**Кодовый замок.** Кодовый замок состоит из панели с кнопками (цифры «0»...«9», кнопка «Вызов», кнопка «Контроль»), цифрового дисплея, электромеханического замка, кнопки «Открыть», звонка. Панель устанавливается с наружной стороны двери, замок устанавливается с внутренней стороны двери, звонок и кнопка «Открыть» устанавливаются внутри охраняемого помещения. В обычном состоянии замок закрыт. Доступ в помещение осуществляется после набора кода доступа, состоящего из четырех цифр. Во время набора кода введенные цифры отображаются на дисплее. Если код набран правильно, то замок открывается на некоторое время, после чего дверь снова закрывается. Содержимое дисплея очищается. Кнопка «Вызов» используется для подачи звукового сигнала внутри помещения. Услышав сигнал, люди внутри помещения могут открыть замок, нажав на кнопку «Открыть». Кнопка «Контроль» используется для смены кодов. Смена кода доступа осуществляется при закрытой двери следующим образом. Нужно набрать код контроля, состоящий из четырех цифр, нажать кнопку «Контроль», и ввести новый код доступа. Для смены кода контроля нужно проделать тоже самое при открытой двери.

Постройте **диаграмму классов** для задачи «Кодовый замок». Один из разработанных классов должен отвечать за логику работы кодового замка. **Этот класс разработайте полностью, на языке Java**

### Вариант 8

**Система составления расписания занятий.** Система обеспечивает составление расписания некоторого учебного заведения, внесение в него изменений, выдачу полного расписания и дополнительной информации (например, по итоговому расписанию составляется расписание указанной группы или указанного преподавателя на заданный день или неделю). В расписании фиксируются время и место проведения занятия, предмет и преподаватель, проводящий занятие, а также номер группы (групп), для которой это занятие проводится.

Некоторые занятия проводятся для нескольких групп: потока или всего курса. Расписание не должно содержать коллизий (например, разные занятия не должны пересекаться друг с другом по месту и времени их проведения, один преподаватель не может вести одновременно два разных занятия, в одно и то же время, у одной и той же группы не допускается одновременно два различных занятия и т. д.).

Система по требованию пользователя должна осуществлять поиск свободных аудиторий на заданную дату и время.

Постройте **диаграмму прецедентов** и **диаграмму классов** для «Системы составления расписания занятий». Диаграмму классов снабдите вашими комментариями.

### Вариант 9

**На языке Java опишите контейнерный класс «Колода карт»,** включающий в себя объекты класса «Игральная карта». В каждом объекте класса «Карта» хранятся масть и достоинство карты. Предусмотрите инициализацию с проверкой допустимости значений полей. В случае недопустимых значений полей выбрасываются исключения. Обеспечьте возможность вывода карт по их масти и достоинству, вывода всех карт колоды, перемешивания колоды и выдачи всех карт из колоды поодиночке и по 6 штук в случайном порядке.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Колода карт» и «Игральная карта», а также отношения между ними.

### Вариант 10

**Домофон.** Домофон регулирует доступ в подъезд многоквартирного дома. В подъезде имеется дверь с замком. С наружной стороны двери установлена панель с кнопками на каждую квартиру, микрофон и динамик. В каждой квартире имеется кнопка «СВЯЗЬ», «БЛОКИРОВКА» и «ОТКРЫТЬ». Кроме того, в квартире имеется микрофон и динамик.

Жильцы могут открывать дверь ключом. Посетитель может нажать кнопку квартиры. В квартире раздается звонок. При нажатии кнопки «СВЯЗЬ» внутри квартиры устанавливается звуковое сообщение между квартирой и посетителем. Звук, произносимые посетителем в микрофон, установленный в подъезде, должны быть слышны в динамике, установленном в квартире. При нажатии на кнопку «ОТКРЫТЬ» дверь в подъезд открывается. Нажатие на кнопку «БЛОКИРОВКА» отключает подачу звонка в квартиру. Повторное нажатие на кнопку «БЛОКИРОВКА» вновь включает подачу звонка. Сведения о посещении дома (когда и из какой квартиры открывали замок домофона) сохраняются в журнале домофона. По прошествии суток журнал домофона пересылается на пульт начальника охраны. После пересылки данных журнал обнуляется.

Постройте **диаграмму прецедентов** и **диаграмму классов** для предметной области «Домофон».

### Вариант 11

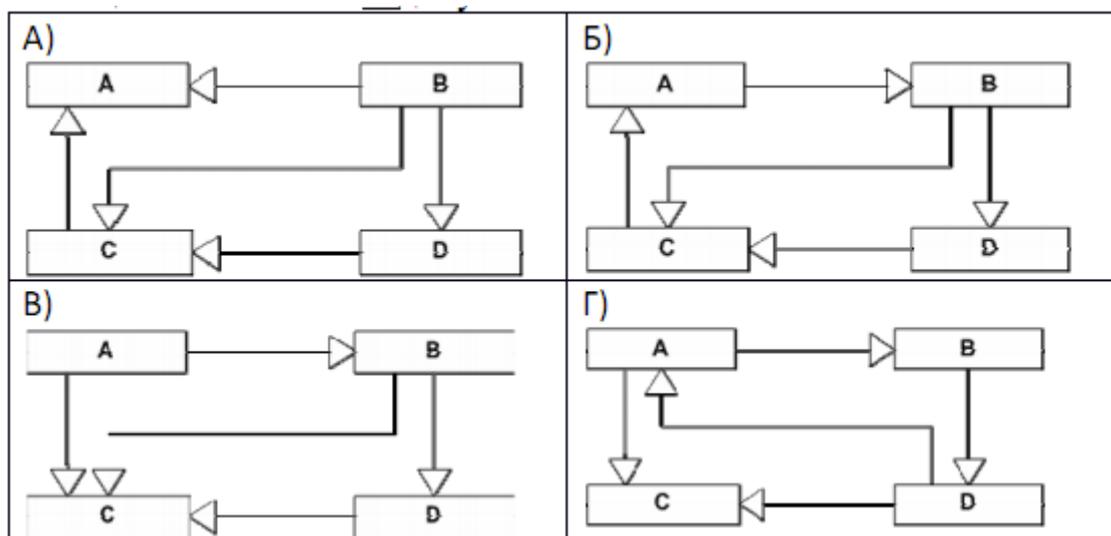
1. Каждый студент университета консультируется у преподавателя, множество преподавателей консультирует множество студентов, некоторые преподаватели не являются консультантами. **Постройте диаграмму классов**, на которой представьте отношения между студентами и преподавателями.
2. **На языке Java опишите класс для работы со строкой**, позволяющей хранить только двоичное число и выполнять с ним арифметические операции. Предусмотрите инициализацию объекта с проверкой допустимости значений. В случае недопустимых значений выбрасываются исключения. **Напишите код, который демонстрирует все разработанные элементы класса.**

### Вариант 12

1. Объект А посылает объекту В несколько сообщений. Приведите примеры UML-диаграмм, на которых будет отображен данный факт.
2. На языке Java опишите класс, представляющий круг. Предусмотрите методы для создания объектов, вычисления площади круга, длины окружности и проверки попадания заданной точки внутрь круга. Опишите методы для получения состояния объекта. Напишите код, который демонстрирует все разработанные элементы класса.

### Вариант 13

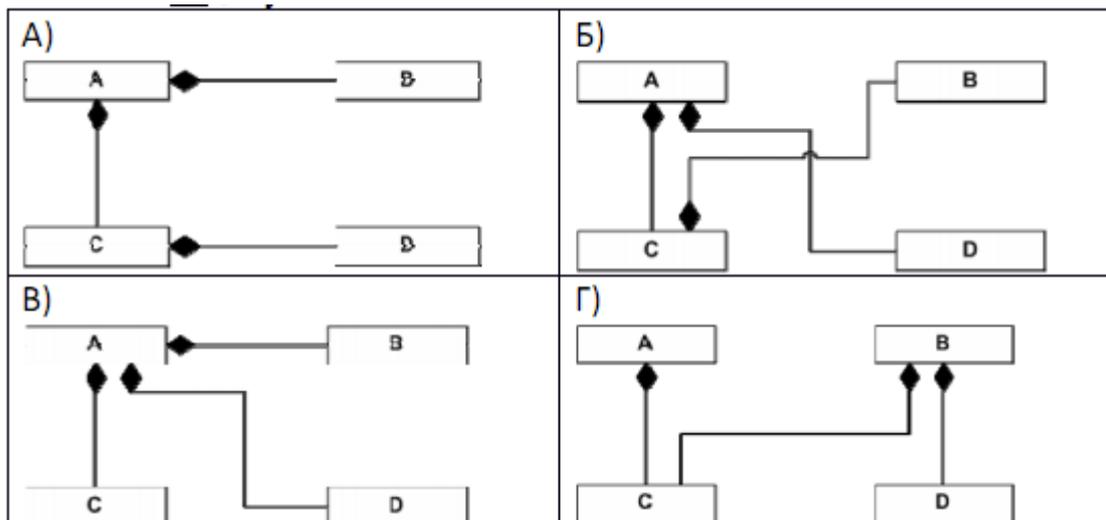
1. А, В, С, D - классификаторы. Какие из показанных диаграмм являются некорректными и почему?



2. На языке Java опишите класс, представляющий треугольник. Предусмотрите методы для создания объектов, вычисления площади и периметра треугольника. Опишите методы для задания и получения состояния объекта. При невозможности построения треугольника выбрасывается исключение. Напишите код, который демонстрирует все разработанные элементы класса.

### Вариант 14

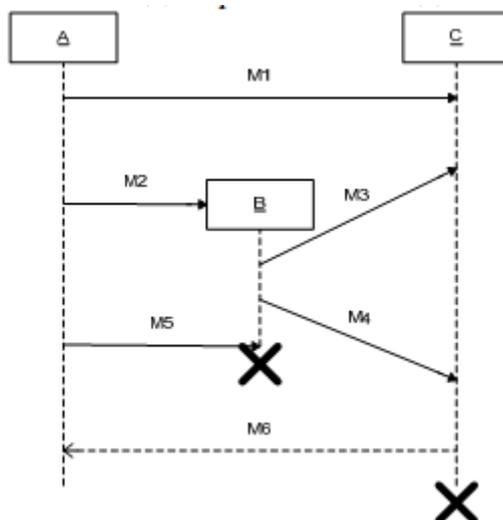
1. А, В, С, D - классификаторы. Какие из показанных диаграмм являются некорректными и почему?



2. На языке Java опишите класс для представления времени. Предусмотрите возможности установки времени и изменения его отдельных полей (час, минута, секунда) с проверкой допустимости вводимых значений. В случае недопустимых значений полей выбрасываются исключения. Создайте методы изменения времени на заданное количество часов, минут и секунд. Напишите код, который демонстрирует все разработанные элементы класса.

### Вариант 15

1. Какие сообщения являются недопустимыми на следующей диаграмме последовательности?



2. На языке Java опишите класс для представления календарной даты. Предусмотрите возможности установки даты и изменения ее отдельных полей (год, месяц, день) с проверкой допустимости вводимых значений. В случае недопустимых значений полей выбрасываются исключения. Создайте методы изменения даты на заданное количество дней, месяцев и лет. Напишите код, который демонстрирует все разработанные элементы класса.

### Вариант 16

**На языке Java опишите контейнерный класс «Студенческая группа»** для хранения сведений о студентах. Для каждого объекта класса «Студент» хранятся его ФИО, дата рождения, номер зачётной книжки, признак присутствия на занятиях. Обеспечьте возможность поиска студентов по разным критериям, вывода списка присутствующих и отсутствующих на занятиях студентов, доступа к сведениям о студенте по его ФИО и номеру зачётной книжки.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Студенческая группа» и «Студент», а также отношения между ними.

### Вариант 17

**На языке Java опишите класс прямоугольников** со сторонами, параллельными осям координат. Предусмотрите возможности создания прямоугольников с проверкой допустимости вводимых значений. В случае недопустимых значений полей выбрасываются исключения. Предусмотрите возможность отображения прямоугольников на экране, перемещения прямоугольников по экрану, изменение их размеров, построение наименьшего прямоугольника, содержащего два заданных прямоугольника, и построение прямоугольника, являющегося общей частью (пересечением) двух прямоугольников.

**Напишите код, который демонстрирует все разработанные элементы класса.**

### Вариант 18

**На языке Java опишите контейнерный класс «Домашняя библиотека»** для хранения сведений о книгах. Для каждого объекта класса «Книга» хранятся её выходные издательские данные, а также признак наличие книги в библиотеке в данный момент (т.е. книга может временно отсутствовать в библиотеке – дали почитать). Обеспечьте возможность поиска книг по разным критериям. Предусмотрите возможность работы с произвольным числом книг, добавления книг в библиотеку, удаления книг из нее, вывода списка присутствующих и отсутствующих книг.

Кроме кода на Java, разработайте **диаграмму классов**, на которой присутствуют классы «Домашняя библиотека» и «Книга», а также отношения между ними.

### Критерии оценки:

100 - балл выставляется студенту, если выполнено 100% задания.

75 - балл выставляется студенту, если выполнено 75% задания.

60 - балл выставляется студенту, если выполнено 60% задания.

Неудовлетворительный балл выставляется студенту, если выполнено менее чем 60% задания.

Составитель \_\_\_\_\_ В.П. Тельнов

(подпись)

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Кафедра Компьютерных систем, сетей и технологий

## Темы групповых и/или индивидуальных творческих заданий/проектов

по дисциплине «Технология разработки программного обеспечения»

### Лабораторная работа №1. (25 вариантов заданий)

Постановка задачи. Разработайте в CASE-средстве *Visual Paradigm* спецификацию требований и технический проект программного обеспечения для предметной области, которую выберите из ПРИЛОЖЕНИЯ на следующей странице.

CASE-средства для разработки: *Visual Paradigm, Netbeans IDE.*

Для развертывания: *web-сервер GlassFish, сервер баз данных MySQL в составе XAMPP.*

#### Рекомендации.

1. Начните работу с изучения рекомендованной литературы и примеров разработок в *Visual Paradigm*.
2. Сформулируйте требования к программному обеспечению (*Requirements Capturing*).
3. Разработайте диаграммы прецедентов (*Use Case Diagram*) для вашего задания.
4. Создайте сценарии выполнения для каждого базового прецедента.
5. Разработайте диаграммы классов (*Class Diagram*) и прочие необходимые *UML*-диаграммы для вашего задания.
6. Сгенерируйте код на языках *C#* и *Java*. Изучите и проанализируйте, каким образом отношения между классами на *UML*-диаграммах трансформировались в код.
7. Подготовьте письменный отчет. Возникающие затруднения попытайтесь преодолеть самостоятельно, потом обращайтесь за помощью.

Письменный отчет по работе должен содержать следующие разделы:

1. Постановку задачи.
2. Спецификацию требований к программному обеспечению, включая:
  - функциональные и прочие существенные требования;
  - диаграмму прецедентов;
  - сценарии выполнения прецедентов.
3. Технический проект, включающий:
  - разработанные *UML*-диаграммы (прецедентов, классов, последовательности, деятельности, компонентов, коммуникации, развертывания);

- текстовые пояснения к каждой из разработанных *UML*-диаграмм (состав, назначение элементов диаграммы, отношения между элементами);
- иерархию классов системы, описание пакетов. Для каждого класса дать краткое описание: ответственность класса; описание атрибутов в виде таблицы из 3-х столбцов (имя, тип, описание); таблицу с описанием операций (полная сигнатура, описание).

4. Листинг сгенерированного кода на языках *C#* и *Java*, выводы по работе.

Для успешной сдачи лабораторной работы необходимо представить письменный отчет, продемонстрировать на практике приемы работы с *CASE*-средствами и ответить на вопросы преподавателя.

**Избегайте представлять к сдаче заимствованные материалы.**

**Работайте самостоятельно!**

**1. \*Видеопрокат\***

Система управления данными о видеокассетах и дисках, об их выдаче клиентам.

Поддерживаемые данные

\* Клиенты

o ФИО

o Контактная информация: адрес, телефон

o Какие носители с фильмами, когда и по какой цене ему выдавались, когда он их возвращал

\* Фильмы

o Название

o Компания, режиссер, год выхода

o Носители (кассеты, диски) и стоимость проката каждого типа носителя

o Количество экземпляров на каждом типе носителя и свободных экземпляров

o Для каждого экземпляра: кому и когда его выдавали, когда он возвращался

Поддерживаемые операции

\* Получение списка клиентов и фильмов

\* Получение истории выдачи и приема фильмов у клиента, списка находящихся у него фильмов

\* Получение истории выдачи и приема экземпляров фильма, сводных сведений о наличии, выдаче и приеме фильмов за заданный интервал времени

\* Внесение информации о выдаче фильма клиенту, получении от него и оплате

\* Добавление и удаление клиента, чтение и редактирование данных о нем

\* Добавление и удаление фильма и отдельных экземпляров, чтение и редактирование данных о фильмах и их экземплярах

## **2. \*Система информации об авиарейсах и билетах\***

Система управления информацией о рейсах, наличии билетов и ценах на них, а также продажи билетов

Поддерживаемые данные

\* Авиарейсы

- o Компания, номер
- o Аэропорты вылета и прилета
- o Время и даты вылета и прилета
- o Стоимость билетов
- o Количество мест и наличие свободных мест

\* Клиенты

- o ФИО
- o Контактная информация: адрес, телефон, e-mail
- o Заказанные билеты, оплаченные билеты
- o Наличие бонусных карт авиакомпаний
- o Налетанные километры по каждой авиакомпании, их использование для оплаты других билетов

Поддерживаемые операции

- \* Получение списка авиарейсов по датам и направлениям, информации о ценах билетов и наличии свободных мест
- \* Получение списка клиентов, в т.ч. летавших определенным рейсом, любыми рейсами авиакомпании, заказавших и оплативших билеты
- \* Получение истории заказов клиента, информации о его бонусах и их использовании
- \* Заказ и оплата билетов на выбранный рейс
- \* Добавление и удаление рейса, чтение и редактирование данных о нем
- \* Добавление и удаление клиента, чтение и редактирование данных о нем

### **3. \*Система информации об автобусных рейсах и билетах\***

Система управления информацией об автобусных рейсах, наличии билетов и ценах на них, а также продажи билетов. Поддерживаемые данные

\* Рейсы

- o Компания, номер
- o Пункты убытия и прибытия, промежуточные остановки
- o Время и даты всех остановок
- o Стоимость билетов для всех пар остановок
- o Количество мест и наличие свободных мест с учетом промежуточных остановок

\* Клиенты

- o ФИО
- o Контактная информация: адрес, телефон, e-mail
- o Заказанные билеты

Поддерживаемые операции

- \* Получение списка рейсов по датам, направлениям и промежуточным остановкам, информации о ценах билетов и наличии свободных мест
- \* Получение списка клиентов, в т.ч. ехавших определенным рейсом, любыми рейсами компании, заказавших билеты
- \* Получение истории заказов клиента
- \* Заказ билетов на выбранный рейс между выбранными пунктами
- \* Добавление и удаление рейса, чтение и редактирование данных о нем
- \* Добавление и удаление клиента, чтение и редактирование данных о нем

#### **4. \*Театральная касса\***

Система учета данных о представлениях и продажи билетов на них.

Поддерживаемые данные

\* Театры

o Режиссеры, актеры

o Адрес

o Количество мест в зале разных видов: партер, балконы, бельэтаж

o Представления

\* Представления

o Театр, режиссер, участвующие актеры

o Даты и время проведения (может быть несколько)

o Продолжительность

o Информация о свободных местах разных видов

o Стоимость билетов разных видов

Поддерживаемые операции

\* Получение списка театров и представлений по театру, режиссеру, занятым актерам, датам проведения

\* Получение данных о наличии свободных мест и стоимости билетов разных видов на представление

\* Покупка билетов

\* Добавление и удаление театра, чтение и редактирование данных о нем

\* Добавление и удаление спектакля, чтение и редактирование данных о нем

## **5. \*Система информации о спортивных соревнованиях\***

Система учета данных о спортивных соревнованиях и продажи билетов на них.

Поддерживаемые данные

\* Соревнования

- o Вид спорта (футбол, синхронное плавание, фигурное катание, гимнастика и пр.)
- o Название, турнир, частью которого оно является
- o Место и время проведения
- o Участвующие (в зависимости от вида спорта): команды и отдельные спортсмены
- o Количество мест в зале разных видов: передние ряды, средние ряды, задние ряды
- o Заказанные и свободные места (для еще не состоявшихся)
- o Результаты (для уже состоявшихся): счет или очки, распределение мест

\* Спортсмены

- o ФИО, возраст
- o История участия в командах и соревнованиях

\* Команды

- o Название
- o Тренеры
- o Состав
- o История участия в соревнованиях

Поддерживаемые операции

- \* Получение списка соревнований по видам спорта, участникам, местам и времени проведения
- \* Получение данных о наличии свободных мест и стоимости билетов разных видов на представление
- \* Покупка билетов
- \* Добавление и удаление соревнования, чтение и редактирование данных о нем
- \* Добавление и удаление команд и спортсменов, чтение и редактирование данных о них

## **6. \*Интернет-магазин бытовой техники\***

Система учета данных о товарах и заказах.

Поддерживаемые данные

**\* Товары**

o Вид (телевизоры, DVD-проигрыватели, холодильники, стиральные машины и пр.)

o Цена

o Компания-производитель, место сборки

o Характеристики, в зависимости от вида (телевизор — габариты, диагональ, разрешение, формат экрана, количество каналов, и пр.; холодильник — габариты, цвет, одно/двухкамерный, расположение камер, мин. температура в морозильнике, объем камер, энергопотребление и пр.; стиральная машина — габариты, макс. загрузка, обороты, энергопотребление и пр.)

o Наличие, количество

**\* Клиенты**

o ФИО

o Контактная информация: адрес, телефон, e-mail

o Сделанные заказы

**\* Заказы**

o Дата и время

o Клиент

o Товары и их количество, общая стоимость

o Условия доставки (адрес, время — определяются клиентом)

o Текущий статус: в обработке, собран, поставлен

Поддерживаемые операции

**\* Получение списка товаров по типам, производителям и характеристикам**

**\* Получение данных о характеристиках, наличии и цене товара**

**\* Оформление заказа**

**\* Проверка статуса заказа**

**\* Добавление и удаление клиента, чтение и редактирование данных о нем**

**\* Добавление и удаление товара, чтение и редактирование данных о нем**

\*

\*

## **7. \*Книжный Интернет-магазин\***

Система учета данных о клиентах, книгах и заказах на них.

Поддерживаемые данные

\* Книги

- o Название
- o Авторы
- o Жанр
- o Издательство, год издания, количество страниц, вид обложки
- o Цена
- o Наличие, количество

\* Клиенты

- o ФИО
- o Контактная информация: адрес, телефон, e-mail
- o Сделанные заказы

\* Заказы

- o Дата и время
- o Клиент
- o Товары и их количество, общая стоимость
- o Условия доставки (адрес, время — определяются клиентом)
- o Текущий статус: в обработке, собран, поставлен

Поддерживаемые операции

- \* Получение списка книг по жанрам, авторам и др. характеристикам
- \* Получение данных о наличии и цене книг
- \* Оформление заказа, проверка и изменение статуса заказа
- \* Добавление и удаление клиента, чтение и редактирование данных о нем
- \* Добавление и удаление книги, чтение и редактирование данных о ней

## **8. \*Информационная система автосалона\***

Система учета данных о клиентах, автомобилях и заказах.

Поддерживаемые данные

\* Автомобили

- o Марка
- o Производитель
- o Регистрационный номер
- o Технические характеристики (объем и мощность двигателя, расход топлива, количество дверей, мест, вместимость багажника, автоматическая коробка передач, круиз-контроль, требуемое топливо и т.п.)
- o Встроенные устройства (кондиционер, радио, видео, GPS-навигатор и пр.)
- o Потребительские характеристики (обивка салона, цвет и пр.)
- o Изменяемые характеристики (пробег, последнее ТО и др.)
- o Цена
- o Клиенты, проводившие тест-драйв

\* Клиенты

- o ФИО
- o Контактная информация: адрес, телефон, e-mail
- o Сделанные заказы

\* Заказы

- o Дата и время
- o Клиент
- o Характеристики автомобиля
- o Нужен ли предварительный тест-драйв
- o Текущий статус: в обработке, ожидание поставки, есть в салоне, в тест-драйве, выполнен

Поддерживаемые операции

- \* Получение списка автомобилей по разным характеристикам
- \* Получение списка клиентов по характеристикам их заказов
- \* Оформление заказа, проверка и изменение статуса заказа
- \* Добавление и удаление клиента, чтение и редактирование данных о нем
- \* Добавление и удаление марки автомобилей или конкретного автомобиля, чтение и редактирование данных о них

## **9. \*Складской учет\***

Система учета данных о товарах на складе, поставщиках и потребителях.

Поддерживаемые данные

\* Товары

- o Наименование
- o Вид (продукты, бытовая химия, одежда-обувь, бытовая электроника)
- o Характеристики, в зависимости от вида: габариты, срок хранения
- o Поставщики и потребители
- o Наличие, количество и единицы его измерения
- o Время хранения и статус (для портящихся)
- o Место хранения (номера помещения и полки)

\* Поставщики и потребители

- o Наименование
- o Контактная информация: адрес(а), телефон(ы), e-mail(ы)
- o Сделанные поставки и заказы

\* Поставки и выдачи

- o Дата и время
- o Поставщик (для поставок) или потребитель (для выдач)
- o Товары и их количество

Поддерживаемые операции

- \* Получение списка имеющихся товаров по видам, сроку хранения, поставщику и пр.
- \* Получение данных о поставках и выдачах за заданный период времени
- \* Оформление поставки или выдачи
- \* Проверка наличия свободного места для поставки
- \* Добавление и удаление товара, чтение и редактирование данных о нем
- \* Добавление и удаление поставщиков и потребителей, чтение и редактирование данных о них

## **10. \*Учебное расписание\***

Система составления расписаний и ведения данных об учебных курсах в ВУЗе.

Поддерживаемые данные

- \* Студенты
  - o ФИО
  - o Год обучения, поток, группа
  - o Какие курсы и когда посещал
- \* Преподаватели
  - o ФИО
  - o Проводимые курсы (ранее и теперь)
- \* Аудитории
  - o Номер
  - o Вместимость
- \* Курсы
  - o Название
  - o Охват: поток, группа, спец. курс
  - o Интенсивность (сколько пар в неделю)
  - o Год обучения (для обязательных)
- \* Занятия
  - o Курс, преподаватель
  - o Аудитория
  - o Время
  - o Студенты

Поддерживаемые операции

- \* Получение списков студентов по потокам и группам
- \* Получение списков преподавателей, в т.ч. по проводимым курсам
- \* Получение списков аудиторий, свободных в определенном интервале
- \* Получение расписания на заданный интервал времени для студента, преподавателя или аудитории
- \* Составление расписания занятий для курса на семестр
- \* Добавление и удаление студентов и преподавателей, чтение и редактирование данных о них, занесение студента в список слушателей спец. курса
- \* Добавление и удаление курса, чтение и редактирование данных о нем

## **11. \*Учебный центр\***

Система составления расписаний и ведения данных об учебных курсах в тренинговом центре.

Поддерживаемые данные

- \* Обучающиеся
  - o ФИО
  - o Посещаемые курсы
- \* Компании
  - o Название, адрес
  - o Проводимые курсы
  - o Преподаватели
- \* Преподаватели
  - o ФИО
  - o Компания
  - o Проводимые курсы
- \* Курсы
  - o Время — день, несколько дней, две недели, месяц
  - o Интенсивность (сколько часов в день)
- \* Занятия
  - o Курс, преподаватель
  - o Время
  - o Обучающиеся

Поддерживаемые операции

- \* Получение списков обучающихся по курсам, истории обучения для данного человека
- \* Получение списков преподавателей, в т.ч. по проводимым курсам
- \* Получение расписания на заданный интервал времени для обучающегося, преподавателя
- \* Составление расписания занятий для курса
- \* Добавление и удаление обучающихся и преподавателей, чтение и редактирование данных о них, занесение обучающегося в список слушателей курса
- \* Добавление и удаление курса, чтение и редактирование данных о нем

## **12. \*Система информации о персонале компании\***

Система управления информацией о персонале.

Поддерживаемые данные

\* Служащие

- o ФИО
- o Домашний адрес
- o Образование
- o Срок работы в компании
- o История занимаемых должностей

\* Должности

- o Название
- o Обязанности

\* Подразделения

- o Название
- o Руководитель
- o Должности (с количеством позиций) и занимающие их люди
- o Внутренние подразделения
- o Головное подразделение

Поддерживаемые операции

- \* Получение списка подразделений, структуры подразделений
- \* Получение списка служащих, в т.ч. по подразделениям, по сроку работы, по должностям
- \* Получение истории для данного служащего
- \* Назначение служащего на новую должность в заданном подразделении
- \* Добавление и удаление служащего, чтение и редактирование данных о нем
- \* Добавление и удаление подразделения или должности, чтение и редактирование данных о них

### **13. \*Кадровое агентство\***

Система управления информацией о вакансиях и резюме.

Поддерживаемые данные

\* Люди

o ФИО

o Домашний адрес

o Образование

o История работы: компании, должности, зарплаты

o Статус: ищет работу или нет, если ищет, какие условия  
(должность, зарплата)

\* Компании

o Название

o Вакансии: должность + предлагаемая зарплата + требования к  
образованию и послужному списку

Поддерживаемые операции

\* Получение списка резюме по образованию, компаниям, в которых  
люди работали, по занимавшимся должностям, зарплатам

\* Получение списка вакансий по компаниям, должностям, зарплатам

\* Получение истории работы для данного человека

\* Поиск подходящих вакансий на резюме и подходящих резюме на вакансию

\* Добавление и удаление данных о человеке, чтение и редактирование  
данных о нем, добавление данных о новом трудоустройстве

\* Добавление и удаление компании, чтение и редактирование данных о  
них, добавление, удаление и редактирование вакансий

#### **14. \*Зарплатная ведомость\***

Система управления информацией о зарплатах служащих компании.

Поддерживаемые данные

\* Служащие

o ФИО

o Домашний адрес

o Дата рождения

o Образование

o Стаж работы в компании

o Текущая должность

o Участие в проектах и выполняемые роли

o История занимаемых должностей и участия в проектах проектов

o Общая история всех выплат

o Премии и даты их выписки

\* Проекты

o Название, даты начал и окончания

o Роли в проекте (руководитель, аналитик, секретарь, эксперт)

\* Политики выплат

o По должностям

o По проектам и ролям

o За стаж

o Премииальные на Новый год, дни рождения, круглые даты в истории компании

Поддерживаемые операции

\* Получение списка служащих, в т.ч. по должностям, проектам, стажу, премированных и пр.

\* Получение истории участия в проектах и карьерной истории для служащего

\* Получение истории выплат для служащего

\* Назначение служащего на новую должность, добавление в/удаление из проекта

\* Добавление и удаление служащего, чтение и редактирование данных о нем

\* Добавление и удаление проекта, чтение и редактирование данных о нем

\* Добавление и удаление политик выплат, чтение и редактирование данных о них.

## **15. \*Клиентская база юридической фирмы\***

Система управления информацией о клиентах и оказываемых им услугах.

Поддерживаемые данные

- \* Клиенты – организации и физические лица
  - o Наименование или ФИО
  - o Контакты: контактные лица, адрес(а), телефон(ы), e-mail(ы)
  - o История услуг: услуги, в какое время оказывались, кто из служащих был задействован
- \* Служащие
  - o ФИО
  - o Домашний адрес, телефон(ы), e-mail(ы)
  - o Образование, должность
  - o История работы: участие в оказании услуг
- \* Услуги
  - o Наименование (создание, восстановление и сопровождение документов, банкротство, эмиссия акций, сопровождение сделок, судебное представительство, консультации)
  - o Стоимость

Поддерживаемые операции

- \* Получение списка клиентов, в т.ч. по оказываемым услугам в заданном интервале времени, задействованным служащим и пр.
- \* Получение списка служащих по их участию в оказании услуг заданным клиентам и в заданное время
- \* Регистрация договора об оказании услуги
- \* Добавление и удаление данных о клиенте, чтение и редактирование данных о нем
- \* Добавление и удаление служащего, чтение и редактирование данных о нем

## **16. \*Биллинговая база оператора связи\***

Система управления информацией о клиентах, оказываемых им услугах и об оплате услуг.

### Поддерживаемые данные

- \* Клиенты – физические лица и организации
  - o Наименование или ФИО
  - o Контакты: контактные лица, адрес(а), телефон(ы), e-mail(ы)
  - o История услуг: услуги, в какое время оказывались
- \* Услуги
  - o Наименование
  - o Характеристики: номер, группа номеров, Интернет, SMS, спец. предложения
  - o Тарифный план (какая часть услуги в какое время сколько будет стоить)
- \* Счета клиентов
  - o Баланс
  - o Поступления на счет
  - o Списания за оказание услуг связи
  - o Ограничения: размер максимального кредита и сроки его погашения

### Поддерживаемые операции

- \* Получение списка клиентов, в т.ч. по оказываемым услугам в заданном интервале времени, по характеристикам их счетов
- \* Получение росписи операций по счету клиента за заданный интервал времени
- \* Регистрация договора об оказании услуги
- \* Регистрация поступлений на счета списаний
- \* Добавление и удаление данных о клиенте, чтение и редактирование данных о нем
- \* Добавление и удаление услуги, чтение и редактирование данных о ней

## **17. \*Система информации о счетах клиентов банка\***

Поддерживаемые данные

\* Отделения

- o Название
- o Адрес
- o Клиенты и счета

\* Клиенты – физ. лица и организации

- o Наименование или ФИО
- o Контакты: контактные лица, адрес(а), телефон(ы), e-mail(ы)
- o Счета

\* Счета

- o Номер
- o Клиент
- o Текущий баланс
- o Вид счета
- o Отделение
- o Начисления/списания

\* Виды счетов

- o Наименование
- o Максимальный кредит и ограничения на его погашение
- o Доходность, интервал и метод выплаты процентов (на этот же счет, на другой)
- o Возможности списания/начисления и ограничения на списываемые/начисляемые суммы

Поддерживаемые операции:

- \* Получение списка клиентов, в т.ч. по типам, видам счетов в заданном интервале времени и пр.
- \* Получение списка счетов по их видам, списаниям/начислениям за заданный период
- \* Получение списка отделений банка, в т.ч. по клиентам, счетам и пр.
- \* Оформление списания/начисления, включая автоматический учет процентов
- \* Заведение счета и его закрытие, чтение данных о нем
- \* Добавление и удаление данных о клиенте, чтение и редактирование данных о нем
- \* Добавление и удаление отделения банка, чтение и редактирование данных о нем

## **18. \*Библиотека\***

Система библиотечного учета для управления данными о читателях и книгах, о выдаче книг читателям.

Поддерживаемые данные

\* Читатели

- o ФИО
- o Номер читательского билета
- o Контактная информация: адрес, телефон
- o Какие книги и когда ему выдавались, когда он их возвращал

\* Книги

- o Название
- o Авторы
- o Издательство, год издания, ISBN
- o Количество экземпляров в библиотеке и свободных экземпляров
- o Для каждого экземпляра: кому и когда его выдавали, когда он возвращался

Поддерживаемые операции

- \* Получение списка читателей и книг, книг - по авторам, названиям и издательствам
- \* Получение истории выдачи и приема книг у читателя, списка находящихся у него книг
- \* Получение истории выдачи и приема экземпляров книги, сводных сведений о наличии, выдаче и приеме книг за заданный интервал времени
- \* Внесение информации о выдаче книг читателю и получении от него
- \* Добавление и удаление читателя, чтение и редактирование данных о нем
- \* Добавление и удаление книги и отдельных экземпляров, чтение и редактирование данных о книгах и их экземплярах

## **19. \*Web-\*\*форум\***

Система управления информацией об обсуждениях на форуме.

Поддерживаемые данные

- \* Пользователь
  - o Login/пароль
  - o Дата регистрации
  - o Права — пользователь или модератор
- \* Тема
  - o Раздел форума
  - o Сообщения по теме
    - + Пользователь, приславший сообщение
    - + Заголовок сообщения
    - + Дата и время поступления
    - + Прикрепленные файлы

Поддерживаемые операции

- \* Получение списка пользователей, в т.ч. по участию в различных разделах и по активности (количеству сообщений в заданном интервале времени)
- \* Получение списка разделов, тем в разделе, сообщений в теме
- \* Для модераторов: создание/удаление раздела, удаление тем, сообщений, создание и блокирование пользователей
- \* Для обычных пользователей: создание тем, создание сообщений в теме

## **20. \*Система генеалогической информации\***

Система управления информацией о родственных связях людей.

Поддерживаемые данные

\* Человек

- o Полное имя
- o Даты рождения и смерти
- o Краткая характеристика — кто это такой, чем занимался(ется)
- o Места проживания
- o Родители
- o Супруги и даты брака и развода (если был развод)
- o Дети от разных браков и внебрачные

Поддерживаемые операции

- \* Получение списка людей по фамилиям, разнообразным родственным связям с определенным человеком (родители, дети, супруги, братья-сестры, родственники во втором колене, по супругам и пр).
- \* Получение генеалогического дерева человека — все предки
- \* Получение дерева потомков человека
- \* Получение всех видов родственных связей между двумя людьми
- \* Добавление данных о человеке, их чтение и редактирование

## **21. \*Система информации о структуре собственности\***

Система управления информацией о структуре собственности для некоторой группы компаний.

Поддерживаемые данные

\* Физические лица

- o ФИО
- o Краткая биография
- o Собственность — в каких компаниях каким процентом акций владеет

\* Компании

- o Название
- o Действует/потеряла статус отдельного юр. лица
- o Год основания
- o История смены названий
- o История сделок по покупке/поглощению других компаний
- o Владельцы — кто или какая компания какой частью акций владеет
- o Владения — в каких компаниях какой частью владеет

Поддерживаемые операции

- \* Получение списка людей по прямо или непрямо контролируемым ими компаниям
- \* Получение полной информации о собственности для человека или компании — каким процентом где владеют, с транзитивным замыканием
- \* Получение полной структуры владения для компании — кто и какой частью владеет, с транзитивным замыканием
- \* Получение цепочки связи между двумя компаниями (как направленной, как и со сменой направления владения)
- \* Добавление данных о человеке или компании, их чтение и редактирование

## **22. \*Астрономический каталог\***

Система управления информацией об астрономических объектах и явлениях.

Поддерживаемые данные

### \* Объекты

- o Класса: звезда (в т.ч. кратная), туманность, галактика, планета, малая планета, спутник, астероид, комета, метеорный поток
- o Тип в классе: для звезд — цвет и пр., для галактик — форма, и т.д.
- o Имена и идентификаторы по разным каталогам
- o Дата открытия
- o Первооткрыватель
- o Характеристики для неподвижных (относительно звезд) объектов: координаты, созвездие, светимость, масса, расстояние от Солнца
- o Характеристики для подвижных: параметры орбиты, вариации скорости движения, масса, изменения светимости
- o Связанные явления

### \* Явления

- o Вид: прохождение, покрытие, затмение, соединение, противостояние, прохождение апоцентра и перицентра, вспышка, столкновение и пр.
- o Связанные объекты и их роли
- o Время начала и конца

Поддерживаемые операции

- \* Получение списка объектов по типам и др. характеристикам, по связанным явлениям в заданном интервале времени, в заданной области неба
- \* Получение списка явлений по объектам, в заданном интервале времени, в заданной области неба
- \* Добавление данных об объекте или явлении, их чтение и редактирование

### **23. \*Коллекция минералов\***

Система управления данными о минералогической коллекции.

Поддерживаемые данные

**\* Минералы**

- o Название
- o Классификация (раздел, класс, подкласс) (см. Wikipedia)
- o Состояние (жидкое, газообразное, аморфное, кристаллическое)
- o Для твердых - тип кристаллической решетки, твердость, хрупкость
- o Блеск, цвет, магнитные свойства
- o Химическая формула
- o Происхождение (осадочное, вулканическое, метаморфическое)
- o Имеющиеся образцы

**\* Образцы**

- o Входящие минералы и способ их включения (кристаллы, вкрапления, примерная % часть образца)
- o Возможное происхождение (метеорит, извержение, осадочные слои и пр.)
- o Место обнаружения (координаты и описание, например, обрыв на правом берегу реки Камы)
- o Источник (экспедиция, дар, обмен с другими коллекциями, пр.)

**\* Экспедиции**

- o Даты начала и конца
- o Участники
- o Собранные образцы и места сбора

Поддерживаемые операции

- \* Получение списка образцов по минералам, источникам и др. характеристикам
- \* Добавление данных о минерале, их чтение и редактирование
- \* Добавление данных об экспедициях и образцах, их чтение и редактирование

## **24. \*Информационная система заповедника\***

Система управления данными о животных в заповеднике.

Поддерживаемые данные

### **\* Животные**

- o Классификация: тип, класс, семейство, вид, латинское название
- o Персональный идентификатор или имя
- o Устанавливавшиеся метки (кольца, RFID и пр.), их идентификаторы, время установки и снятия, кто устанавливал
- o Особенности внешнего вида
- o Особенности поведения
- o Статус: мигрирующее, постоянно в заповеднике; живое или уже нет
- o Связи с другими животными: родители, потомки, текущее положение в группе/стае, текущий партнер
- o История болезней: болезнь, время фиксации болезни, время фиксации выздоровления, кто и какую помощь оказывал, последствия

### **\* Работник заповедника**

- o ФИО
- o Образование
- o Стаж работы
- o С животными каких видов работал

Поддерживаемые операции

- \* Получение списка животных по видам, имеющимся или прошлым меткам, перенесенным болезням
- \* Получение списка сотрудников по образованию и опыту работы, с какими животными имел дело
- \* Получение деталей по животному, истории его меток, истории его болезней, связей с другими
- \* Добавление данных о животном или работнике, их чтение и редактирование

## **25. \*Агентство недвижимости\***

Система управления информацией о предложениях и заказах в агентстве недвижимости.

Поддерживаемые данные

### **\* Заказы**

- о Контактная информация клиента
- о Вид сделки: аренда, покупка, обмен
- о Требования (с ограничениями, точными значениями или без ограничений)
  - + объект: комната/квартира/дом
  - + тип дома: деревянный/панельный/кирпичный/монолит
  - + площадь: общая/жилая, комнат, кухни, прихожей
  - + наличие и площадь лоджии/балкона
  - + наличие удобств: отдельный/совмещенный санузел, внешние удобства, электричество, газ, канализация, водопровод, телефон, телевидение, Интернет
  - + этаж
  - + состояние: новостройка/нет, время после последнего ремонта
  - + расстояние до метро, МКАД, ближайшей ж/д станции, остановки автобуса/троллейбуса/трамвая
  - + расположение: округ/район Москвы, район/город Московской обл.
  - + максимальная цена

### **о Предложения**

- о Контактная информация
- о Вид сделки
- о Характеристики (те же, что в заказах, с точными значениями, кроме адреса и цены)
- о Адрес
- о Начальная цена

Поддерживаемые операции

- \* Получение списка заказов/предложений по различным характеристикам
- \* Поиск подходящих предложений на заказ и заказов на предложение
- \* Добавление и удаление данных о заказе или предложении, чтение и редактирование данных о них

## Лабораторная работа №2. Реализация программного решения

### Постановка задачи.

Реализуйте (т.е. воплотите в коде) программное решение, технический проект которого вы разработали в Лабораторной работе №1. Платформа разработки: *J2EE* (язык *Java*).

CASE-средства для разработки: *Visual Paradigm, Netbeans IDE*.

Для развертывания: *web*-сервер *GlassFish*, сервер баз данных *MySQL* в составе *XAMPP*.

При реализации программного обеспечения применяйте компонентные технологии.

В ходе работы, возможно, потребуется внесение изменений в первоначальный технический проект. Вносите и документируйте эти изменения.

Добейтесь того, чтобы ваше программное решение функционировало в полном соответствии со спецификацией требований и *UML*-диаграммами технического проекта.

Снабдите программное решение оконным графическим интерфейсом и средствами взаимодействия с базами данных (при необходимости).

Выполните «обратное проектирование», получив из разработанного кода диаграмму классов. Сопоставьте её с первоначально разработанной вами диаграммой классов и сделайте необходимые выводы.

Письменный отчет по работе должен содержать следующие разделы:

5. Постановку задачи.
6. Диаграмму компонентов и диаграмму развертывания программного решения.
7. Эскизы окон графического интерфейса программного решения.
8. Надлежаще оформленный (структурированный, комментированный) листинг разработанного авторского кода.
9. Протокол изменений, внесенных в первоначальный технический проект.
10. Результаты «обратного проектирования», выводы по работе.

Для успешной сдачи лабораторной работы необходимо представить письменный отчет, продемонстрировать на практике приемы работы с *CASE*-средствами, работоспособность программного решения (в том числе при работе в сети) и ответить на вопросы преподавателя.

## Лабораторная работа №3 Тестирование и рефакторинг

### Постановка задачи.

Выполните комплексное автоматизированное тестирование программного решения, которое вы разработали в Лабораторных работах № 1 и 2.

Для этого продумайте и разработайте сценарии тестирования, тестовые наборы данных, драйверы тестирования.

Выполните следующие виды тестирования в режиме «черного ящика» и «белого ящика»:

1. Модульное (*unit*-тестирование).
2. Интеграционное тестирование.
3. Системное тестирование.

Выполните расчет метрик тестирования (не менее 3-х метрик, на выбор). Метрики процесса тестирования по стандарту IEEE 982.1-1998 см. в теме «Тестирование софта» на сайте <http://ksst.leadhoster.com/>.

Продумайте варианты улучшения внешнего вида кода, компоновки папок проекта, оптимизации программного решения (рефакторинг). Выполните рефакторинг проекта средствами *Netbeans IDE*.

Платформа разработки: *J2EE* (язык *Java*).

CASE-средства для разработки: *Visual Paradigm, Netbeans IDE*.

Для развертывания: *web*-сервер *GlassFish*, сервер баз данных *MySQL* в составе *XAMPP*.

Письменный отчет по работе должен содержать следующие разделы:

11. Постановку задачи.
12. Сценарии тестирования, тестовые условия и наборы данных.
13. Надлежаще оформленный (структурированный, комментированный) листинг кода разработанных драйверов тестирования.
14. Протоколы тестирования в табличной форме. Выявленные и устраненные ошибки.
15. Результаты тестирования (включая метрики), рефакторинга и оптимизации кода, выводы по работе.

Для успешной сдачи лабораторной работы необходимо представить письменный отчет, продемонстрировать работу программного решения под управлением драйверов тестирования и ответить на вопросы преподавателя.

## Лабораторная работа №4. Документирование и презентация

### Постановка задачи.

Выполните документирование и презентацию программного обеспечения, которое вы разработали и протестировали в Лабораторных работах № 1-3.

Для этого продумайте и создайте демонстрационный пример, который максимально полно покажет возможности разработанного вами программного обеспечения.

В соответствии с действующими стандартами, подготовьте эксплуатационную документацию для программного обеспечения, минимум три позиции из ниже перечисленного:

- ❖ руководства и инструкции (User Guide, User Manual);
- ❖ справки по функциям продукта (Reference);
- ❖ ответы на вопросы (Frequently Asked Questions);
- ❖ решение известных проблем (Troubleshooting);

❖ статьи How-To.

Подготовьте итоговую презентацию (файл ppt) разработанного программного обеспечения, в которую включите все существенные материалы Лабораторных работ № 1-3. Объем презентации: 15-20 слайдов.

Письменный отчет по работе включает:

1. Комплект эксплуатационной документации для программного обеспечения, минимум три позиции из ниже перечисленного:
  - ❖ руководства и инструкции (User Guide, User Manual);
  - ❖ справки по функциям продукта (Reference);
  - ❖ ответы на вопросы (Frequently Asked Questions);
  - ❖ решение известных проблем (Troubleshooting);
  - ❖ статьи How-To.
2. Презентацию (формат *MS Power Point*) разработанного программного обеспечения.

Для успешной сдачи лабораторной работы необходимо представить комплект эксплуатационной документации для программного обеспечения и презентацию в твердой копии (на бумажном носителе); показать работу программного решения на демонстрационном примере (в том числе работу в сети); показать презентацию, ответить на вопросы студентов и преподавателя.

### **Критерии оценки:**

100 - балл выставляется студенту, если выполнено 100% задания.

75 - балл выставляется студенту, если выполнено 75% задания.

60 - балл выставляется студенту, если выполнено 60% задания.

Неудовлетворительный балл выставляется студенту, если выполнено менее чем 60% задания.

Составитель \_\_\_\_\_ В.П. Тельнов

(подпись)

« \_\_\_\_\_ » 20 \_\_\_\_ г.



## ЛИСТ СОГЛАСОВАНИЯ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств разработан в отделении биотехнологий ИАТЭ НИЯУ МИФИ

<p>Рассмотрен на заседании отделения биотехнологий и рекомендован к одобрению Ученым советом ИАТЭ НИЯУ МИФИ</p> <p>(протокол № <u>9/1</u> от «<u>21</u>» <u>04</u> 20<u>23</u> г.)</p>	<p>Начальник отделения биотехнологий ИАТЭ НИЯУ МИФИ</p> <p></p> <p>А.А. Котляров</p>
--	---