

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Б1.В.ДВ.01.01 Паттерны проектирования программного обеспечения

Специальность/направление подготовки: **09.03.01 Информатика и вычислительная техника**

Специализация/направленность(профиль): **Проектирование программного обеспечения**

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

1.1. Цели:

Приобретение базовых знаний и навыков проектирования и разработки приложений с применением объектно-ориентированного подхода и паттернов.

1.2. Задачи:

Оценка необходимости и эффективности применения в разработке крупных программных комплексов требуемых паттернов.

Применять паттерны для разработки на современных объектно-ориентированных языках программирования.

Применения современных методов проектирования программного обеспечения; современных методов оценки качества программного обеспечения.

2. ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ИНДИКАТОРЫ ИХ ДОСТИЖЕНИЯ

ПКС-2 : Способен применять типовые решения и методы проектирования программного обеспечения

ПКС-2.1 : Знает базовые типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методы и средства проектирования программного обеспечения, баз данных, программных интерфейсов

ПКС-2.2 : Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов

ПКС-2.3 : Владеет базовыми навыками проектирования структур данных, баз данных, программных интерфейсов

3. КРАТКАЯ ХАРАКТЕРИСТИКА СОДЕРЖАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Код занятия	Темы, планируемые результаты их освоения	Семестр	Часов	Прак. подг.
1.1	<p>Тема 1. Объектно-ориентированный подход к созданию информационных систем Краткое содержание: Основные определения предмета курса и основные понятия. Сравнительный анализ различных парадигм программирования. Структурное программирование. Процедурное программирование. Объектно-ориентированное программирование. Знать: Основные определения предмета курса и основные понятия. Сравнительный анализ различных парадигм программирования. Структурное программирование. Процедурное программирование. Объектно-ориентированное программирование. /Лек/</p>	7	4	0
1.2	<p>Практическая работа 1. Использование паттернов проектирования при разработке приложений. Краткое содержание: диаграммы вариантов использования, абстрактный вариант использования, ассоциации Уметь: создавать диаграммы вариантов использования, абстрактный вариант использования, Владеть навыками добавления ассоциации, добавления связи расширения, создания главной диаграммы классов. /Пр/</p>	7	4	0
1.3	<p>Тема 1. Объектно-ориентированный подход к созданию информационных систем Краткое содержание: Основные определения предмета курса и основные понятия. Сравнительный анализ различных парадигм программирования. Структурное программирование. Процедурное программирование. Объектно-ориентированное программирование. Знать: Основные определения предмета курса и основные понятия. Сравнительный анализ различных парадигм программирования. Структурное программирование. Процедурное программирование. Объектно-ориентированное программирование. Уметь: создавать диаграммы вариантов использования, абстрактный вариант использования Владеть навыками добавления ассоциации, добавления связи расширения, создания главной диаграммы классов. /Ср/</p>	7	10	0
1.1	<p>Тема 2. Объектно-ориентированный анализ и проектирование Краткое содержание: Декомпозиция. Базовые принципы проектирования. Восходящее и нисходящее проектирование. Объектно-ориентированный подход. Состояние, поведение и идентичность объекта. Принципы объектно-ориентированного проектирования.</p>	7	4	0

	<p>Архитектурные паттерны. Паттерны проектирования. Паттерны анализа. Паттерны тестирования. Паттерны реализации.</p> <p>Объектно-ориентированное моделирование, язык UML Модели в различных областях человеческой деятельности. Роль моделирования в процессе разработки. UML и его назначение. Функциональные требования и диаграммы использования. Система, действующие лица и варианты использования. Включение и расширение. Диаграммы активностей. Моделирование бизнес-процессов: диаграммы BPMN и IDEF0.</p> <p>Диаграммы классов и моделирование предметной области.</p> <p>Моделирование взаимодействия объектов. Способы моделирования свойств. Ассоциации, обобщение, зависимости. Диаграммы компонент и диаграммы размещения. Моделирование данных: ER, UML диаграммы. Диаграммы состояний и их применение для описания поведения реактивных систем. Диаграммы последовательности и взаимодействия, коммуникационные диаграммы.</p> <p>Знать: Базовые принципы проектирования. Объектно-ориентированное моделирование, язык UML. Диаграммы классов и моделирование предметной области. /Лек/</p>			
1.2	<p>Практическая работа 2. Создание структурных диаграмм UML. Создание поведенческих диаграмм UML</p> <p>Краткое содержание: Создание структурных диаграмм UML. Создание поведенческих диаграмм UML</p> <p>Уметь: создавать классы, создавать методы</p> <p>Владеть: навыками адаптировать один класс к потребностям другого класса /Пр/</p>	7	4	0
1.3	<p>Тема 2. Объектно-ориентированный анализ и проектирование</p> <p>Краткое содержание: Декомпозиция. Базовые принципы проектирования. Восходящее и нисходящее проектирование. Объектно-ориентированный подход. Состояние, поведение и идентичность объекта. Принципы объектно-ориентированного проектирования. Архитектурные паттерны. Паттерны проектирования. Паттерны анализа. Паттерны тестирования. Паттерны реализации. Объектно-ориентированное моделирование, язык UML Модели в различных областях человеческой деятельности. Роль моделирования в процессе разработки. UML и его назначение. Функциональные требования и диаграммы использования. Система, действующие лица и варианты использования. Включение и расширение. Диаграммы активностей. Моделирование бизнес-процессов: диаграммы BPMN и IDEF0. Диаграммы классов и моделирование предметной области. Моделирование взаимодействия объектов. Способы моделирования свойств. Ассоциации, обобщение, зависимости. Диаграммы компонент и диаграммы размещения. Моделирование данных: ER, UML диаграммы. Диаграммы состояний и их применение для описания поведения реактивных систем. Диаграммы последовательности и взаимодействия, коммуникационные диаграммы.</p> <p>Знать: Базовые принципы проектирования. Объектно-ориентированное моделирование, язык UML. Диаграммы классов и моделирование предметной области.</p> <p>Уметь: создавать классы, создавать методы</p> <p>Владеть: навыками адаптировать один класс к потребностям другого класса /Ср/</p>	7	10	0
1.1	<p>Тема 3. Шаблоны проектирования</p> <p>Структурные шаблоны. Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Рекурсивная композиция, использование полиморфизма и композиции при едином базовом интерфейсе. Выделение алгоритма в объект, шаблон Strategy. Сокрытие реализации. Проектирование удаленного и отложенного выполнения.</p> <p>Порождающие шаблоны. Паттерн Abstract Factory. Паттерн Builder. Паттерн Factory Method. Паттерн Prototype. Паттерн Singleton. Поведенческие шаблоны. Паттерн Chain of Responsibility. Паттерн Command. Паттерн Interpreter. Паттерн Iterator. Паттерн Mediator.</p> <p>Знать: Структурные шаблоны. Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Порождающие шаблоны. Поведенческие шаблоны. /Лек/</p>	7	4	0
1.2	<p>Практическая работа 3. Использование структурных паттернов и паттернов поведения при разработке приложений.</p> <p>Краткое содержание Структурные шаблоны. Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Поведенческие шаблоны. Паттерн Chain of Responsibility. Паттерн Command. Паттерн Interpreter. Паттерн Iterator. Паттерн Mediator.</p> <p>Уметь работать со структурными шаблонами: Паттерн Adapter. Паттерн Bridge.</p>	7	4	0

	<p>Паттерн Composite. Паттерн Decorator. Владеть навыками работы с поведенческими шаблонами: Паттерн Chain of Responsibility. Паттерн Command. Паттерн Interpreter. Паттерн Iterator. Паттерн Mediator. /Пр/</p>			
1.3	<p>Тема 3. Шаблоны проектирования Структурные шаблоны. Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Рекурсивная композиция, использование полиморфизма и композиции при едином базовом интерфейсе. Выделение алгоритма в объект, шаблон Strategy. Скрытие реализации. Проектирование удаленного и отложенного выполнения. Порождающие шаблоны. Паттерн Abstract Factory. Паттерн Builder. Паттерн Factory Method. Паттерн Prototype. Паттерн Singleton. Поведенческие шаблоны. Паттерн Chain of Responsibility. Паттерн Command. Паттерн Interpreter. Паттерн Iterator. Паттерн Mediator. Знать: Структурные шаблоны. Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Порождающие шаблоны. Поведенческие шаблоны. Уметь работать со структурными шаблонами: Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Владеть навыками работы с поведенческими шаблонами: Паттерн Chain of Responsibility. Паттерн Command. Паттерн Interpreter. Паттерн Iterator. Паттерн Mediator. /Ср/</p>	7	10	0
1.1	<p>Тема 4. Архитектурные шаблоны Краткое содержание: Архитектурные стили. Понятие архитектурного шаблона и стиля. Основные архитектурные шаблоны. Многоуровневая архитектура. Каналы и фильтры. Клиент — сервер. Модель — представление — контроллер. Управляемая событиями архитектура. Архитектура на основе микросервисов. Знать: Архитектурные стили. Понятие архитектурного шаблона и стиля. Основные архитектурные шаблоны. Многоуровневая архитектура. Каналы и фильтры. /Лек/</p>	7	4	0
1.2	<p>Практическая работа 4. Разбор архитектур существующих приложений. Краткое содержание стиль клиент/сервер, одноранговые (Peer-to-Peer, P2P) приложения, серверы приложений Уметь создавать стиль клиент/сервер, одноранговые (Peer-to-Peer, P2P) приложения Владеть навыками использования системы клиент-очередь-клиент, серверов приложений /Пр/</p>	7	4	0
1.3	<p>Тема 4. Архитектурные шаблоны Краткое содержание: Архитектурные стили. Понятие архитектурного шаблона и стиля. Основные архитектурные шаблоны. Многоуровневая архитектура. Каналы и фильтры. Клиент — сервер. Модель — представление — контроллер. Управляемая событиями архитектура. Архитектура на основе микросервисов. Знать: Архитектурные стили. Понятие архитектурного шаблона и стиля. Основные архитектурные шаблоны. Многоуровневая архитектура. Каналы и фильтры. Уметь создавать стиль клиент/сервер, одноранговые (Peer-to-Peer, P2P) приложения Владеть навыками использования системы клиент-очередь-клиент, серверов приложений /Ср/</p>	7	10	0
1.4	<p>Зачет с оценкой Знать: базовые типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методы и средства проектирования программного обеспечения, баз данных, программных интерфейсов Уметь: применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов Владеть: базовыми навыками проектирования структур данных, баз данных, программных интерфейсов /ЗаО/</p>	7	0	0

4. ФОРМА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

ЗаО: 7 семестр

Разработчик программы Гайфуллина Зульфия Альфитовна



И.о. зав. кафедрой Одиноква Е.В.

