Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»
Дата и время: 2024-02-21 00:00:00
471086fad29a3b30e244c728abc3661ab35e9d50310dcf0e75e03a5b6fdf6436

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кемеровский государственный университет»
Новокузнецкий институт (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»

(Наименование филиала, где реализуется данная дисциплина)

Факультет информатики, математики и экономики

Утверждаю
Декан ФИМЭ
_____ А.В. Фомина
«13» февраля 2020 года

Рабочая программа дисциплины

Б1.О.19 Программная инженерия
Код, название дисциплины

Направление подготовки
09.03.03 Прикладная информатика
Код, название направления

Направленность (профиль) подготовки
Прикладная информатика в экономике

Программа бакалавриата

Квалификация выпускника
бакалавр

Форма обучения
Очно-заочная

Год набора 2020

Новокузнецк 2020

Лист внесения изменений

в РПД Б1.О.19 Программная инженерия

(код по учебному плану, название дисциплины)

Сведения об утверждении:

утверждена Ученым советом факультета информатики, математики и экономики (протокол Ученого совета факультета № 8 от 13.02.2020)

для ОПОП 2020 год набора на 2020 / 2021 учебный год

по направлению подготовки 09.03.03 Прикладная информатика

(*код и название направления подготовки / специальности*)

направленность (профиль) Прикладная информатика в экономике

Одобрена на заседании методической комиссии факультета информатики, математики и экономики (протокол методической комиссии факультета № 6 от 06.02.2020)

Одобрена на заседании кафедры информатики и вычислительной техники им. В.К. Буторина

протокол № 6 от 23.01.2020 г. Маркидонов А.В. / _____ _____

(Ф. И.О. зав. кафедрой) (*Подпись)*

Оглавление

1 Цель дисциплины. 5

1.1 Формируемые компетенции 5

1.2 Индикаторы достижения компетенций 5

1.3 Знания, умения, навыки (ЗУВ) по дисциплине 5

2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации. 6

3. Учебно-тематический план и содержание дисциплины. 7

3.1 Учебно-тематический план 7

3.2. Содержание занятий по видам учебной работы 8

4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации. 11

5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины. 12

5.1 Учебная литература 12

5.2 Материально-техническое и программное обеспечение дисциплины 12

5.3 Современные профессиональные базы данных и информационные справочные системы 15

6 Иные сведения и (или) материалы. 15

6.1.Примерные темы письменных учебных работ 15

6.2. Примерные вопросы и задания / задачи для промежуточной аттестации 16

1 Цель дисциплины.

В результате освоения данной дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы бакалавриата (далее - ОПОП):

ОПК-8 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла.

Содержание компетенций как планируемых результатов обучения по дисциплине см. таблицы 1 и 2.

1.1 Формируемые компетенции

Таблица 1 - Формируемые дисциплиной компетенции

Наименование вида компетенции *(универсальная, общепрофессиональная, профессиональная)*	Наименование категории (группы) компетенций	Код и название компетенции
общепрофессиональная		ОПК-8 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла

1.2 Индикаторы достижения компетенций

Таблица 2 – Индикаторы достижения компетенций, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Дисциплины и практики, формирующие компетенцию ОПОП
ОПК-8 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла	ОПК 8.1 Координирует работы по созданию, адаптации и сопровождению информационной системы ОПК 8.2 Организовывает работы по управлению проектом создания информационных систем на стадиях жизненного цикла проекта	**Б1.О.19 Программная инженерия** Б1.О.20 Управление ИТ-проектами Б2.О.03(У) Учебная практика. Технологическая (проектно-технологическая) практика Б3.01(Д) Выполнение и защита выпускной квалификационной работы

1.3 Знания, умения, навыки (ЗУВ) по дисциплине

Таблица 3 – Знания, умения, навыки, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
ОПК-8 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла	ОПК 8.1 Координирует работы по созданию, адаптации и сопровождению информационной системы	Знать: понятие и модели жизненного цикла программных систем; принципы и методы программной инженерии, реинженерии, реверсной инженерии и рефакторинга применительно к программным

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
		системам; технологии, парадигмы и шаблоны проектирования и программирования. Уметь: выбирать и обосновывать технологии, методы и шаблоны проектирования и программирования на всех стадиях жизненного цикла; применять CASE-средства проектирования и программирования на всех стадиях жизненного цикла. Владеть: методами, технологиями и парадигмами проектирования и программирования для создания программных систем; методами обеспечения и оценки качества программных систем; методами и средствами визуального моделирования программных систем

2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации.

Таблица 4 – Объем и трудоемкость дисциплины по видам учебных занятий

Общая трудоемкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения
	ОЗФО
1 Общая трудоемкость дисциплины	180
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	56
Аудиторная работа (всего):	56
в том числе:	14
лекции	
практические занятия, семинары	32
практикумы	
лабораторные работы	
в интерактивной форме	
в электронной форме	
Внеаудиторная работа (всего):	
в том числе, индивидуальная работа обучающихся с преподавателем	
подготовка курсовой работы /контактная работа	
групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем*)*	
творческая работа (эссе)	

3 Самостоятельная работа обучающихся (всего)	98
4 Промежуточная аттестация обучающегося - экзамен и объём часов, выделенный на промежуточную аттестацию:	36 6 семестр

3. Учебно-тематический план и содержание дисциплины.

3.1 Учебно-тематический план

Таблица 5 - Учебно-тематический план очно-заочной формы обучения

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость *(всего час.)*	Грудоемкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			
			Аудиторн. занятия		СРС	
			лекц.	практ.		
Семестр 6						
	1. Базовые понятия программной инженерии					
1-2	1.1 Основные понятия и принципы программной инженерии	16	1	2	13	
3	1.2 Реинженерия, реверсная инженерия, рефакторинг	8	1	2	5	Реферат
4	1.3 Технология конвейерной сборки	8	1	2	5	
	2. Парадигмы программирования				0	
5	2.1 Модульное программирование	8	1	2	5	Индивидуальное задание №1
6	2.2 Парадигма объектного программирования	14	1	2	11	Индивидуальное задание №2
7	2.3 Парадигма компонентного программирования	8	1	2	5	Индивидуальное задание №3
8	2.4 Генерирующее программирование	8	1	2	5	
9	2.5 Сервисное программирование	8	1	2	5	
	3. Технология систем				0	
10	3.1 Проектирование сложных систем	8	1	2	5	
11	3.2 Паттерны и фреймворки в архитектуре ИС	8	1	2	5	Реферат
12	3.3 Моделирование доменов средствами онтологии	8	1	2	5	
13	3.4 Обеспечение качества программных систем	8	1	2	5	Индивидуальное задание №4
14	3.5 Тестирование и экспертирование программных систем	8	1	2	5	Индивидуальное задание №5
	4. Визуальное моделирование программного обеспечения				0	
15	4.1 Унифицированный язык моделирования UML	4	1	-	3	
15-18	4.2 Визуальные модели и диаграммы программных средств	8	1	2	5	Индивидуальное задание №6
	Промежуточная аттестация - экзамен	36				экзамен
	Всего:	180	14	32	98	

3.2. Содержание занятий по видам учебной работы

Таблица 6 – Содержание дисциплины

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
	Семестр 6	
Содержание лекционного курса		
1	*Базовые понятия программной инженерии*	
1.1	Сущность и методы программной инженерии	Основные понятия программной инженерии. Элементы и процессы. Программа. Метод разработки. Управление разработкой. Жизненный цикл. Модель жизненного цикла.
1.1	Принципы программной инженерии	Принцип производственно организации. Принцип обеспечения технологичности. Принцип планирования трудозатрат.
1.2	Реинженерия, реверсная инженерия, рефакторинг	Понятия и методы реинженерии, реверсной инженерии. Задачи реверсной инженерии. Метод рефакторинга
1.3	Технология конвейерной сборки	Сущность сборочного конвейера. Линии программ. Product Lines. Метод сборки специализированных технологий
2	*Парадигмы программирования*	
2.1	Модульное программирование	Понятие модуля и интерфейса. Методы сборки. Теория сборки разнородных модулей. Фундаментальные типы данных. Простые и сложные типы данных. Общие типы данных. Неструктурые и генерированные типы данных. Стили сборочного программирования. Матричное представление графов из модулей. Отношение достижимости модулей графов. Операции построения модульных структур. Отладка и тестирование модулей. CASE-средства интеграции модулей и интерфейсов.
2.2	Парадигма объектного программирования	Математическое моделирование объектной модели. Алгебра объектного анализа предметной области. Методы объектов и их интерфейсы. Жизненный цикл объектного моделирования предметной области. CASE-средства объектного подхода в современных средах
2.3	Парадигма компонентного программирования	Теория компонентного программирования. Базовые понятия. Модели разработки систем из компонентов. Операции внешней, внутренней и эволюционной алгебры. Типизация компонентов. Корректность сборки компонентов. Жизненный цикл компонентной разработки программных средств. CASE-средства поддержки компонентов и систем
2.4	Генерирующее программирование	Элементы программных систем и семейств систем. Трансформация и конфигурация программных систем. Аспектно-ориентированное программирование. Модели конструирования вариантных систем и семейств. Модели сложных и распределенных систем. CASE-системы поддержки мультипрограммирования
2.5	Сервисное программирование	Сервис. Сервисная и компонентно-ориентированные архитектуры. Сервисы контрактов WCF

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
3	*Технология систем*	
3.1	Проектирование сложных систем	Базовые подходы к проектированию сложных систем. Генерация и сборка сложных систем. Методология проектирования систем с помощью жизненного цикла. Модели и методы проектирования вариантов систем
3.2	Паттерны и фреймворки в архитектуре ИС	Шаблон проектирования. Архитектурные паттерны. Системные паттерны. Структурные паттерны. Поведенческие паттерны. Порождающие паттерны. Паттерны параллельного программирования. Антипаттерны. Фреймворки.
3.3	Моделирование доменов средствами онтологии	Онтологическое моделирование проблемной области. Описание доменов методами онтологии. Основные понятия онтологии представления предметной области. Формализация онтологической модели жизненного цикла. Онтология процесса тестирования жизненного цикла.
3.4	Обеспечение качества программных систем	Основные задачи проблемы управления качеством. Моделирование характеристик качества программных средств. Задачи управления качеством программных средств. Модель требований с ориентацией на обеспечение качества программных средств. Система прогнозирования безотказной работы программных средств. Анализ достижения уровня качества. Задачи оценки качества сложных систем. Эталонная модель качества оценки показателей программных средств.
3.5	Тестирование и экспертирование программных систем	Модель тестирования и определение оптимального времени. Экспертирование компонентов и систем. Методы управления программным проектом.
4	*Визуальное моделирование программного обеспечения*	
4.1	Унифицированный язык моделирования UML	Графические нотации моделирования. Принципы моделирования. Составляющие UML. Элементы нотации. Направления унификации. Область применения UML
4.2	Визуальное моделирование программных средств	Диаграммы вариантов использования. Диаграммы последовательности. Диаграммы кооперации. Диаграммы классов. Диаграммы состояний. Диаграммы размещения.
4.2	Инструментальные средства построения	Технологии и инструментальные средства. Унифицированный процесс разработки.
Содержание практических занятий		
1	*Базовые понятия программной инженерии*	
1.1	Выбор методов разработки. Составление моделей.	Анализ моделей жизненного цикла. Описание предметной области. Представление системы в виде модели
1.1	Построение модели жизненного цикла	Модели жизненного цикла. События. Принципы.
1.2	Применение методов реверсной инженерии, реинженерии и рефакторинга	Анализ и перестройка структуры программного кода и программных систем

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
1.3	Сборка систем в .NET	Сборка, взаимодействие программ в разных средах. Механизмы передачи данных.
2	*Парадигмы программирования*	
2.1	Реализация методов сборки разнородных модулей	Процессы практической реализации сборки разнородных модулей
2.2	Математическое моделирование объектной модели	Проектирование модели предметной области на четырех уровнях логико-математического определения объектов.
2.2	Методы объектов и их интерфейсы	CASE-средства объектного подхода
2.3	Применение парадигмы компонентного программирования	Типизация компонент. Разработка программы с использованием компонентного подхода
2.4	Составление моделей программных систем	Трансформационная модель. Конфигурационная модель. Аспектно-ориентированное программирование. Модель взаимодействия систем.
2.5	Механизмы сервисного программирования	Сервисная и компонентно-ориентированная архитектура.
3	*Технология систем*	
3.1	Технология сборки сложных систем	Генерация и сборки сложных систем
3.2	Применение и реализация паттернов	Решение практических задач с использованием паттернов
3.3	Построение онтологической модели	Описание доменов методами онтологии.
3.4	Анализ достижения уровня качества	Решение задач оценки качества сложных систем
3.5	Построение модели тестирования программных средств	Определение модели тестирования и оптимального времени тестирования.
4	*Визуальное моделирование программного обеспечения*	
4.2	Построение диаграмм использования и диаграмм последовательности	Использование UML
4.2	Построение диаграмм классов	Использование UML
4.2	Построение диаграмм состояний и диаграмм размещение	Использование UML
4.2	Реализация программной системы на основе визуальной модели	Реализация программных систем с использованием модели и паттернов
	Промежуточная аттестация - *экзамен*	

4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.

Для положительной оценки по результатам освоения дисциплины обучающемуся необходимо выполнить все установленные виды учебной работы. Оценка результатов работы обучающегося в баллах (по видам) приведена в таблице 7.

Таблица 7 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам (БРС)

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (17 недель)
Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий)	**60**	Лекционные занятия (конспект) (17 занятий)	**0,5 балла** посещение 1 лекционного занятия	0 – 8,5
		Лабораторные работы (отчет о выполнении лабораторной работы) (18 работ).	**1 балл** - посещение 1 практического занятия и выполнение работы на 51-65% **1,25 балла** – посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85,1-100%	18 – 22,5
		Индивидуальные задания (отчет о выполнении индивидуального задания) (6 работ)	**За одну КР от 2 до:** **3 балла** (выполнено 66 - 85% заданий) **4 балла** (выполнено 86 - 100% заданий)	12 - 24
		Реферат (по разделу 1 или 3 по выбору)	**1 балла** (пороговое значение) **5 баллов** (максимальное значение)	1 - 5
Итого по текущей работе в семестре				51 - 100
Промежуточная аттестация (экзамен)	40 (100% /баллов приведенной шкалы)	Теоретический вопрос 1.	**3 балла** (пороговое значение) **6 баллов** (максимальное значение)	3 - 6
		Теоретический вопрос 2.	**3 балла** (пороговое значение) **6 баллов** (максимальное значение)	3 - 6
		Теоретический вопрос 3.	**3 балла** (пороговое значение) **6 баллов** (максимальное значение)	3 - 6
		Решение задачи 1.	**5,5 балла** (пороговое значение) **11 баллов** (максимальное значение)	5,5 - 11
		Решение задачи 2.	**5,5 балла** (пороговое значение) **11 баллов** (максимальное значение)	5,5 - 11
Итого по промежуточной аттестации (экзамену)				(51 – 100% по приведенной шкале) 20 – 40 б.
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины.

5.1 Учебная литература

Основная учебная литература

Лаврищева, Е. М. Программная инженерия. Парадигмы, технологии и case-средства: учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. — Москва : Издательство Юрайт, 2019. — 280 с. — (Университеты России). — ISBN 978-5-534-01056-5. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://biblio-online.ru/bcode/444952 (дата обращения: 02.12.2019)

Дополнительная учебная литература

Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для академического бакалавриата / Е. А. Черткова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 147 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-09172-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://biblio-online.ru/bcode/437536 (дата обращения: 02.12.2019).

Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 432 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-07604-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://biblio-online.ru/bcode/436514 (дата обращения: 02.12.2019).

Рыбальченко, М. В. Архитектура информационных систем : учебное пособие для вузов / М. В. Рыбальченко. — Москва : Издательство Юрайт, 2019. — 91 с. — (Университеты России). — ISBN 978-5-534-01159-3. — URL: https://biblio-online.ru/bcode/437686 (дата обращения: 18.02.2020).— Текст : электронный

5.2 Материально-техническое и программное обеспечение дисциплины

Учебные занятия по дисциплине проводятся в учебных аудиториях НФИ КемГУ

Таблица 8 – Материально-техническое и программное обеспечение дисциплины

Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы	Перечень основного оборудования, учебно-наглядных пособий и используемого программного обеспечения	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом

716 Учебная аудитория для проведения: - занятий лекционного типа; Специализированная (учебная) мебель: доска меловая, кафедра, столы, стулья.	Оборудование для презентации учебного материала:переносные - ноутбук, экран, проектор. Используемое программное обеспечение: MSWindows (MicrosoftImaginePremium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), FoxitReader (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер(отечественное свободно распространяемое ПО). Интернет с обеспечением доступа в ЭИОС.	654079, Кемеровская область, г. Новокузнецк, пр-кт Металлургов, д. 19
501 Компьютерный класс. Учебная аудитория (мультимедийная) для проведения: - занятий семинарского (практического) типа; - групповых и индивидуальных консультаций; - текущего контроля и промежуточной аттестации;	Специализированная (учебная) мебель: доска меловая, кафедра, столы компьютерные, стулья. Оборудование для презентации учебного материала:стационарное - компьютер преподавателя, экран, проектор. Оборудование: стационарное - компьютеры для обучающихся (17 шт.). Используемое программное обеспечение: MSWindows (MicrosoftImaginePremium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), BloodshedDevC++ 4.9.9.2 (свободно распространяемое ПО), FoxitReader (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Java	654079, Кемеровская область, г. Новокузнецк, пр-кт Металлургов, д. 19

	(бесплатная версия), Qt (свободно распространяемое ПО), Eclipse (свободно распространяемое ПО), MicrosoftVisualStudio (MicrosoftImaginePremium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.) Интернет с обеспечением доступа в ЭИОС.	

5.3 Современные профессиональные базы данных и информационные справочные системы.

Перечень СПБД и ИСС по дисциплине

CITForum.ru - on-line библиотека свободно доступных материа-лов по информационным технологиям на русском языке - http://citforum.ru

Научная электронная библиотека eLIBRARY.RU – крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты - www.elibrary.ru

Единое окно доступа к образовательным ресурсам - http://window.edu.ru/

6 Иные сведения и (или) материалы.

6.1.Примерные темы письменных учебных работ

Темы рефератов

Раздел 1

1. Задачи анализа системы для проведения изменений в структуре кода
2. Задачи расширения функциональности программных средств
3. Задачи замены платформы и языка программирования
4. Задачи изменения моделей и структур данных
5. Задачи восстановления структуры системы и компонентов
6. Задачи выбора подходящего языка программирования
7. Развитие реверсной инженерии
8. Операции рефакторинга

Раздел 3

9. Применение архитектурных паттернов в проектирование программных систем
10. Применение системного паттерна Модель-Вид-Контроллер
11. Примеры применения системного паттерна Обратный вызов
12. Примеры применения структурного паттерна Адаптер
13. Примеры применения структурного паттерна «Мост»
14. Примеры применения структурного паттерна Компоновщик
15. Примеры применения структурного паттерна «Декоратор»
16. Примеры применения структурного паттерна «Фасад»
17. Примеры применения структурного паттерна «Приспособленец»
18. Примеры применения структурного паттерна Цепочка ответственности
19. Примеры применения структурного паттерна Команда
20. Примеры применения структурного паттерна Интерпритатор
21. Примеры применения структурного паттерна Состояние
22. Применение порождающих паттернов в проектировании программных систем
23. Основные паттерны параллельного программирования
24. Антипаттерны в управлении разработкой ПО и их характеристика
25. Антипаттерны в разработке ПО и их характеристика

26. Антипаттерны в объектно-ориентированном программировании и их характеристика
27. Антипаттерны в области программирования и их характеристика
28. Методологические антипаттерны и их характеристика
29. Организационные антипаттерны и их характеристика
30. Фреймворк Захмана

Темы индивидуальных заданий

Индивидуальное задание 1

Составить схему трансформации типов данных в программном продукте, основанном на модульном программировании. Реализовать программный продукт, на основе принципов модульного программирования

Индивидуальное задание 2

1. Составить логико-математическую модель предметной области
2. Описать поведение объекта
3. Описать схему вызова объектов для программного продукта для модели предметной области

Индивидуальное задание 3

Описать и проанализировать задачу, решенную с использованием парадигмы компонентного программирования

Индивидуальное задание 4

Проанализировать программную систему, используя эталонную модель качества оценки показателей

Индивидуальное задание 5

Построить математическую модель оценки объектов предметной области

Индивидуальное задание 6

Используя язык моделирования UML, разработать программное средство, опираясь на анализ предметной области

6.2. Примерные вопросы и задания / задачи для промежуточной аттестации

Таблица 9 - Примерные теоретические вопросы и практические задачи к зачету экзамену

Разделы и темы	Примерные теоретические вопросы	Примерные практические задачи
1. Базовые понятия программной инженерии		
1.1 Основные понятия и принципы программной инженерии	1. Подходы программной инженерии 2. Модели жизненного цикла программных систем 3. Принципы программной инженерии	Определить затраты на стадиях жизненного цикла проекта
1.2 Реинженерия, реверсная инженерия, рефакторинг	4. Процессы реинженерии 5. Задачи реверсной инженерии	Выбрать язык программирования для программного средства
1.3 Технология конвейерной сборки	6. Фабрика программ 7. Сущность сборочного конвейера 8. Метод сборки специализированных технологий	

2. Парадигмы программирования		
2.1 Модульное программирование	9. Теория сборки разнородных модулей. 10. Стили сборочного программирования 11. Операции построения модульных структур	Представить систему модулей
2.2 Парадигма объектного программирования	12. Алгебра объектного анализа предметной области 13. Методы объектов	Построить математическую модель предметной области
2.3 Парадигма компонентного программирования	14. Модели разработки систем из компонентов 15. Типизация компонентов	Провести типизацию компонентов
2.4 Генерирующее программирование	16. Аспектно-ориентированное программирование 17. Трансформация и конфигурация программных систем	Построить модель сложной системы
2.5 Сервисное программирование	18. Сервис 19. Сервисы контрактов WCF	Определить сервисы и архитектуру
3. Технология систем		
3.1 Проектирование сложных систем	20. Генерация и сборка сложных систем 21. Модели и методы проектирования вариантов систем	
3.2 Паттерны и фреймворки в архитектуре ИС	22. Архитектурные паттерны. 23. Антипаттерны.	Для решения задачи из предметной области выбрать наиболее подходящие паттерны
3.3 Моделирование доменов средствами онтологии	24. Описание доменов методами онтологии. 25. Онтология процесса тестирования жизненного цикла.	
3.4 Обеспечение качества программных систем	26. Моделирование характеристик качества программных средств. 27. Система прогнозирования безотказной работы программных средств. 28. Эталонная модель качества оценки показателей программных средств.	Оценить качества показателей на основе эталонной модели качества
3.5 Тестирование и экспертирование программных систем	29. Модель тестирования и определение оптимального времени. 30. Методы управления программным проектом.	Определить оптимальное время тестирование программного средства
4. Визуальное моделирование программного обеспечения		
4.1 Унифицированный язык моделирования UML	31. Область применения UML 32. Графические нотации моделирования.	
4.2 Визуальные модели и диаграммы программных средств	33. Диаграммы вариантов использования. 34. Унифицированный	Построить диаграмму вариантов использования для заданной предметной области

	процесс разработки	Построить диаграмму классов для заданной предметной области

Составитель (и): Штейнбрехер О.А., доцент кафедры информатики и вычислительной техники им. В.К. Буторина

(*фамилия, инициалы и должность преподавателя (ей)*)