

Подписано электронной подписью:  
Вержицкий Данил Григорьевич  
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»  
Дата и время: 2024-04-24 00:00:00

471086fad29a3b30e244c728abc3661ab35c9d150210dcf0e75e03a5b6fdf6436  
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Кемеровский государственный университет»  
Кузбасский гуманитарно-педагогический институт  
Факультет информатики, математики и экономики

УТВЕРЖДАЮ  
Декан ФИМЭ  
А.В. Фомина

### **Рабочая программа дисциплины**

## **К.М.10.01 Программная инженерия**

Направление подготовки

Прикладная информатика

Направленность (профиль) подготовки  
09.03.03 Прикладная информатика в образовании

Программа бакалавриата

Квалификация выпускника  
*бакалавр*

Форма обучения  
*Заочная*

Год набора 2023

Новокузнецк 2024

## ОГЛАВЛЕНИЕ

Оглавление .....	2
1 Цель дисциплины .....	3
1.1 Знания, умения, навыки (ЗУВ) по дисциплине .....	3
2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации. ....	3
3 Учебно-тематический план и содержание дисциплины .....	4
3.1 Учебно-тематический план.....	4
3.2 Содержание занятий по видам учебной работы .....	5
4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.....	6
5 Учебно-методическое обеспечение дисциплины .....	7
5.1 Учебная литература .....	7
5.2 Материально-техническое и программное обеспечение дисциплины.....	8
5.3 Современные профессиональные базы данных и информационные справочные системы.....	8
6 Иные сведения и (или) материалы .....	9
6.1 Примерные вопросы и задания / задачи для промежуточной аттестации .....	9

## 1 ЦЕЛЬ ДИСЦИПЛИНЫ

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

- *ОПК 1*

### 1.1 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
ПК-1 Способен разрабатывать, внедрять и обеспечивать техническую поддержку информационных систем в образовательной сфере	ПК 1.1 Осуществляет разработку, инсталляцию, модификацию и эксплуатацию алгоритмов и программного обеспечения	Знать: - основные понятия и принципы программной инженерии; - технологии проектирования сложных систем; - средства разработки сложных систем. Уметь: - применять средства моделирования и разработки сложных систем при решении профессиональных задач; Владеть: - навыком выбора технологий программирования для создания программных систем; - моделировать и проектировать сложные системы; - применять визуальные средства моделирования.

## 2 ОБЪЁМ И ТРУДОЁМКОСТЬ ДИСЦИПЛИНЫ ПО ВИДАМ УЧЕБНЫХ ЗАНЯТИЙ. ФОРМЫ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ.

Таблица 2 — 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		
	ОФО	ОЗФО	ЗФО
1 Общая трудоемкость дисциплины			180
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)			21
Аудиторная работа (всего):			12
в том числе:			
лекции			6
практические занятия, семинары			
практикумы			
лабораторные работы			6
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)			125
4 Промежуточная аттестация обучающегося экзамен 8 семестр			9

## 3 УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 3.1 Учебно-тематический план

Таблица 3 — Учебно-тематический план очной формы обучения

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего часов)	Трудоёмкость занятий (час.)						Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			ЗФО			
			Аудиторн. занятия	СРС	СРС	Аудиторн. занятия	СРС	СРС	
			лекц.			практ.			
	<b>1. Основные понятия</b>								
1	1.1 Программная инженерия, программный продукт, программный проект	6				2		4	УО
2	1.2 Технологии программирования	6						6	УО
	<b>2. Жизненный цикл программного продукта</b>								
3	2.1 Понятие жизненного цикла программного продукта	8						8	УО
4	2.2 Модели жизненного цикла и процесса разработки программного продукта	8						8	УО
	<b>3. Модели и процессы управления программным проектом</b>								
5	3.1 Процессы программного проекта	8				2		6	УО
6	3.2 Инициация проекта. Планирование проекта	8						8	УО
7	3.3 Исполнение и завершение проекта	8						8	УО
8	3.4 Мониторинг и управление проектом	8						8	УО
	<b>4. Разработка требований</b>								
9	4.1 Определение и виды программных требований	8				2		6	УО
10	4.2 Разработка требований. Работа с требованиями	8					2	6	ИЗ
	<b>5. Проектирование программных систем</b>								
11	5.1 Ключевые вопросы проектирования	8						8	УО
12	5.2 Архитектура программного обеспечения. Архитектурные стили проектирования	8				2		6	УО
13	5.3 Графическое представление	8					2	6	ИЗ

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего часов)	Трудоемкость занятий (час.)						Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			ЗФО			
			Аудиторн. занятия	СРС	СРС	Аудиторн. занятия	СРС	СРС	
лекц.	практ.	лекц.	практ.						
	архитектуры								
14	5.4 Анализ качества и оценка программного дизайна	8						8	УО
	<b>6. Конструирование программного обеспечения</b>								
15	6.1 Основы конструирования. Разработка баз данных	8					2	6	ИЗ
16	6.2 Структурное программирование. Объектно-ориентированное программирование	8					2	6	ИЗ
17	6.3 Шаблоны проектирования	8					2	6	ИЗ
18	6.4 Система управления версиями	8						8	УО
	<b>7. Тестирование программного обеспечения</b>								
19	7.1 Основы тестирования. Виды тестирования	7						7	УО
20	7.2 Работа с ошибками	6						6	УО
	<b>8. Сопровождение программных систем</b>								
21	8.1 Организация и управление процессом сопровождения. Ресурсы, необходимые для сопровождения	6						6	УО
	<b>9. Качество программного обеспечения</b>								
22	9.1 Основы качества программного обеспечения. Метрики и атрибуты качества	8						8	УО
23	9.2 Управление качеством. Надежность как главная составляющая качества	6						6	УО
24	Промежуточная аттестация - экзамен								экзамен
	Всего:	171					8	8	155

### 3.2 Содержание занятий по видам учебной работы

Таблица 4 – Содержание дисциплины

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
<i>Содержание лекционного курса</i>		
1	<b>Основные понятия</b>	
1.1	<i>Программная инженерия, программный продукт, программный проект</i>	Сущность программной инженерии. Особенности процесса промышленной разработки программного обеспечения. Состав документации, сопровождающей процесс разработки и сопровождения программного продукта согласно стандарту IEEE. Какие этапы развития процесса разработки программного продукта.
2	<b>Модели и процессы управления программным проектом</b>	

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
2.1	Процессы программного проекта	Содержание понятий проекта и процесса управления проектом. Процессы управления проектом.
3.	<b>Разработка требований</b>	
	Определение и виды программных требований	Понятие требований к разрабатываемым программным продуктам. Схема структурирования уровней требований. Основные разделы разработки требований в соответствии с ядром знаний SWEBOOK. Классификация требований. Назначение функциональных и эксплуатационных требований.
	<b>Проектирование программных систем</b>	
	Архитектура программного обеспечения. Архитектурные стили проектирования	Определение архитектуры программного обеспечения. Задачи, решаемые разработкой архитектуры приложения. Определение исходных данных для проектирования архитектуры приложения. Задачи и ограничения определяющие цели архитектуры приложения. Характеристики архитектурных стилей проектирования. Совместимость архитектурных стилей.
<i>Содержание практических занятий</i>		
1	<b>Разработка требований</b>	
1.1	Разработка требований. Работа с требованиями	Процесс разработки требований. анализ осуществимости требований. Осуществление извлечения требований. Процесс формирования и анализа требований. Этапы разработки спецификаций требований.
2	<b>Проектирование программных систем</b>	
2.1	<i>Планирование архитектуры программного обеспечения</i>	Определение исходных данных для проектирования архитектуры приложения. Задачи и ограничения определяющие цели архитектуры приложения. Характеристики архитектурных стилей проектирования. Совместимость архитектурных стилей.
3.	<b>Конструирование программного обеспечения</b>	
3.1	Разработка баз данных	Концептуальное (инфологическое) проектирование. Построение ER-диаграммы для концептуальной модели данных. Дatalogическое проектирование. Построение ER-диаграммы для дatalogической модели данных. Физическое проектирование БД.
3.2	Объектно-ориентированное программирование	Основные понятия объектно-ориентированного программирования. Конструктор класса. Основные виды конструкторов. Деструктор класса. Виртуальная функция. Создание классов и объектов на языке программирования.
3.3	Шаблоны проектирования	Выбор и использование порождающих, структурных и поведенческих шаблонов в различных задачах.
<i>Промежуточная аттестация — экзамен</i>		

#### **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 5.

Таблица 5 — 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) (4 занятия)	2 балла — посещение 1 лекционного занятия	4 - 8
		Практические работы (отчет о выполнении практической работы) (4 работы).	6 балл — посещение 1 практического занятия и выполнение работы на 51-65% 13 баллов — посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85,1-100%	24 - 52
<b>Итого по текущей работе в семестре</b>				30 - 60
Промежуточная аттестация (экзамен)	40 (100% /баллов приведенной шкалы)	Теоретический вопрос 1	5 баллов (пороговое значение) 10 баллов (максимальное значение)	5 - 10
		Теоретический вопрос 2	5 баллов (пороговое значение) 10 баллов (максимальное значение)	5 - 10
		Выполнение задания	10 баллов (пороговое значение) 20 баллов (максимальное значение)	10 - 20
<b>Итого по промежуточной аттестации (экзамену)</b>				(51 – 100% по приведенной шкале) 20 – 40 б.
<b>Суммарная оценка по дисциплине:</b> Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

## 5 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.

### 5.1 Учебная литература

#### Основная учебная литература

1. Введение в программную инженерию : Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчев. — Москва : КУРС: ИНФРА-М, 2019. — 336 с. - ISBN 978-5-906923-22-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1035160> (дата обращения: 02.11.2020). – Режим доступа: по подписке.

2. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для вузов / Е. А. Черткова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 147 с. — (Высшее образование). — ISBN 978-5-534-09172-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452749> (дата обращения: 02.11.2020).

#### Дополнительная учебная литература

1. Введение в программную инженерию : Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчев. — Москва : КУРС: ИНФРА-М, 2019. — 336 с. - ISBN 978-5-906923-22-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1035160> (дата обращения: 02.11.2020). – Режим доступа: по подписке.

2. Подбельский, В. В. Программирование. Базовый курс C# : учебник для вузов / В. В. Подбельский. — Москва : Издательство Юрайт, 2020. — 369 с. — (Высшее образование). — ISBN 978-5-534-10616-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450868> (дата обращения: 02.11.2020).

3. 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, 2020. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452137> (дата обращения: 02.11.2020).

## **5.2 Материально-техническое и программное обеспечение дисциплины**

Учебные занятия по дисциплине проводятся в учебных аудиториях НФИ КемГУ:

Программная инженерия	<p>303 Компьютерный класс. Учебная аудитория (мультимедийная) для проведения занятий:</p> <ul style="list-style-type: none"> <li>- занятий лекционного типа;</li> <li>- занятий семинарского (практического) типа.</li> <li>- текущего контроля и промежуточной аттестации</li> </ul> <p>Специализированная (учебная) мебель: доска маркерно-меловая, столы компьютерные, стулья.</p> <p>Оборудование для презентации учебного материала: стационарное - ноутбук преподавателя, экран, проектор.</p> <p>Оборудование: компьютеры для обучающихся (11 шт.).</p> <p>Используемое программное обеспечение: MSWindows (Microsoft Imagine Premium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), Bloodshed DevC++ 4.9.9.2 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Adobe Reader XI (свободно распространяемое ПО), WinDjView (свободно распространяемое ПО).</p> <p>Интернет с обеспечением доступа в ЭИОС.</p>	654027, Кемеровская область - Кузбасс, г. Новокузнецк, пр-кт Пионерский, д.13, пом.2
-----------------------	---	--

## **5.3 Современные профессиональные базы данных и информационные справочные системы.**

### **Перечень СПБД и ИСС по дисциплине**

1. Единый информационно-аналитический портал государственной поддержки инновационного развития бизнеса (АИС «Инновации») режим доступа <http://innovation.gov.ru/>
2. База данных правовых актов «КонсультантПлюс»: комп. справ. правовая система / компания «КонсультантПлюс». — Электрон. прогр.—[Электронный ресурс] Режим доступа: <http://base.consultant.ru>, свободный. – Загл. с экрана.
3. Каталог паттернов проектирования. Режим доступа Режим доступа: <https://refactoring.guru/ru/design-patterns/catalog>.
4. Хабр. Сообщество IT-специалистов. Хаб «Управление разработкой. Планирование, отслеживание и контроль». Режим доступа: [https://habr.com/ru/hub/dev\\_management/](https://habr.com/ru/hub/dev_management/).
5. Хабр. Сообщество IT-специалистов. Хаб «Управление продуктом. Учимся управлять продуктом» Режим доступа: <https://habr.com/ru/hub/productp/>



6. Хабр. Сообщество IT-специалистов. Хаб «Проектирование и рефакторинг. Реорганизация кода». Режим доступа: <https://habr.com/ru/hub/refactoring/>.
7. Хабр. Сообщество IT-специалистов. Хаб «Менеджмент». Режим доступа: <https://habr.com/ru/flows/management/>.

## 6 ИНЫЕ СВЕДЕНИЯ И (ИЛИ) МАТЕРИАЛЫ

### 6.1 Примерные вопросы и задания / задачи для промежуточной аттестации

Таблица 8 - Примерные теоретические вопросы и практические задания к экзамену

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания
<b>1. Основные понятия</b>		
1.1 Программная инженерия, программный продукт, программный проект	Сущность программной инженерии. Особенности процесса промышленной разработки программного обеспечения. Состав документации, сопровождающей процесс разработки и сопровождения программного продукта согласно стандарту IEEE. Какие этапы развития процесса разработки программного продукта.	
1.2 Технологии программирования	Основные характеристики стихийного подхода к разработке ПО. Основные характеристики структурного подхода к разработке ПО. Основные положения объектного подхода к разработке ПО? Основные характеристики и положения компонентного подхода к разработке ПО.	
<b>2. Жизненный цикл программного продукта</b>		
2.1 Понятие жизненного цикла программного продукта	Понятия жизненного цикла ПП, модели и методологии жизненного цикла. Основные проблемы практического применения модели жизненного цикла. Структура и организацию стандарта жизненного цикла ISO 12207. Основные результаты разработки данного стандарта. Стандарт жизненного цикла ISO 15504. Его отличие от ISO 12207.	
2.2 Модели жизненного цикла и процесса разработки программного	Основные особенности каскадной модели жизненного цикла.	

продукта	<p>Основные особенности спиральной модели. Сходства и различия спиральной модели и классического жизненного цикла.</p> <p>Основные особенности итерационной модели жизненного цикла.</p> <p>Основные особенности инкрементной модели жизненного цикла.</p> <p>Основные промышленные модели процесса разработки программного продукта?</p> <p>Основные фазы модели MSF.</p> <p>Модель RUP и ее основное отличие от модели MSF.</p> <p>Особенности модели жизненного цикла экстремального программирования.</p>	
<b>3. Модели и процессы управления программным проектом</b>		
3.1 Процессы программного проекта	<p>Содержание понятий проекта и процесса управления проектом.</p> <p>Процессы управления проектом.</p>	
3.2 Инициация проекта. Планирование проекта	<p>Понятие инициации проекта.</p> <p>Процессы инициации.</p> <p>Возможные ключевые участники и заинтересованные стороны проекта.</p> <p>Суть стоимостной оценки проекта. Определение сроков реализации проекта.</p> <p>Сущность планирования проекта.</p> <p>Основные процессы планирования проекта.</p> <p>Планирование управления содержанием проекта.</p> <p>Планирование организационной структуры проекта.</p> <p>Управление конфигурацией проекта.</p> <p>Планированием управления качеством проекта.</p>	<p>Проанализировать влияние стоимости необходимых ресурсов на стоимость реализации проекта.</p> <p>Создать иерархическую структуру работы над проектом.</p> <p>Создать расписание проекта</p>
3.3. Исполнение и завершение проекта	<p>Сущность руководства и управления исполнением проекта. Основные процессы исполнения и завершения проекта.</p> <p>Общее планирование проекта.</p> <p>Рабочее планирование проекта.</p> <p>Принципы количественного управления проектом.</p>	
3.4 Мониторинг и управление проектом	<p>Суть мониторинга и управления работами проекта.</p> <p>Основные процессы мониторинга и управления проектом.</p> <p>Содержание процесса мониторинга и контроля рисков.</p> <p>Содержание процесса контроля стоимости проекта.</p> <p>Содержание процесса контроля и управление качеством проекта?</p>	<p>По приведенным данным построить диаграмму разброса.</p> <p>По приведенным данным построить диаграмму тренда.</p> <p>По приведенным данным построить диаграмму Паретто.</p>

<b>4. Разработка требований</b>		
4.1 Определение и виды программных требований	<p>Понятие требований к разрабатываемым программным продуктам. Схема структурирования уровней требований.</p> <p>Основные разделы разработки требований в соответствии с ядром знаний SWEBOK.</p> <p>Классификация требований.</p> <p>Назначение функциональных и эксплуатационных требований.</p> <p>Количественные показатели эксплуатационных требований.</p>	
4.2 Разработка требований. Работа с требованиями	<p>Процесс разработки требований. анализ осуществимости требований.</p> <p>Осуществление извлечения требований.</p> <p>Процесс формирования и анализа требований.</p> <p>Этапы разработки спецификаций требований. Аттестация требований.</p> <p>Итеративная природа работы с требованиями. Процесс управления изменениями требований.</p> <p>Классификация изменяемых требований. Трассировка требований.</p>	<p>На основе записи спецификации требований на естественном языке создать описание требований на основе стандарта ISO/IEC/IEEE 29148:2011.</p> <p>На основе записи спецификации требований на естественном языке создать описание требований на основе Rational Unified Process.</p> <p>На основе записи спецификации требований на естественном языке создать описание пользовательских сценариев.</p>
<b>5. Проектирование программных систем</b>		
5.1 Ключевые вопросы проектирования	<p>Назначение проекта программной системы.</p> <p>Основные цели проектирования.</p> <p>Процесс проектирования?</p> <p>Роли, участвующие в процессе проектирования, их основные задачи.</p>	
5.2 Архитектура программного обеспечения. Архитектурные стили проектирования	<p>Определение архитектуры программного обеспечения.</p> <p>Задачи, решаемые разработкой архитектуры приложения.</p> <p>Определение исходных данных для проектирования архитектуры приложения.</p> <p>Задачи и ограничения определяющие цели архитектуры приложения.</p> <p>Характеристики архитектурных стилей проектирования.</p> <p>Совместимость архитектурных стилей.</p> <p>Методы обеспечения отказоустойчивости, их достоинства и недостатки.</p>	<p>На основе описания требований к программной системе выберите архитектуру для её реализации и обоснуйте выбор.</p>
5.3 Графическое представление архитектуры	<p>Способы графического отображения архитектуры программного средства.</p> <p>Основные средства отображения архитектуры программного</p>	<p>Создайте графическое представление архитектуры программной системы на основе её описания на естественном языке.</p>

	средства.	
5.4 Анализ качества и оценка программного дизайна	Основные атрибуты анализа качества программного дизайна.	
<b>6. Конструирование программного обеспечения</b>		
6.1 Основы конструирования. Разработка баз данных	Содержание процесс конструирования программного обеспечения. Особенности процесса конструирования для основных категории программного обеспечения. Понятие БД. Иерархическая, сетевая, реляционная, объектно-ориентированная, объектно-реляционная БД. Основные понятия реляционной модели базы данных. Компоненты реляционной модели данных. Задачи проектирования базы данных. Концептуальное (инфологическое) проектирование. Принцип построения ER-диаграммы для концептуальной модели данных. Понятие даталогического проектирования. Принципы построения ER-диаграммы для даталогической модели данных. Понятие физического проектирования БД. Особенности конструирования логики работы с данными для СУБД. Понятие безопасности БД. Основные варианты обеспечения защиты данных.	По описанию предметной области осуществить концептуальное проектирование базы данных. На основании инфологического описания базы данных выполнить её даталогическое проектирование для реляционной модели данных. На основании даталогического описания базы данных реализовать её физическую модель для заданной СУБД.
6.2 Структурное программирование. Объектно-ориентированное программирование	Основные преимущества и недостатки структурного программирования. Основные конструкции структурного программирования. Методика разработки программ с использованием структурного программирования. Определение объектно-ориентированного программирования. Основные понятия объектно-ориентированного программирования. Конструктор класса. Основные виды конструкторов. Деструктор класса. Виртуальная функция.	На основе реализации классов постройте диаграмму классов в нотации UML. Напишите реализацию класса на основе описания его функциональности. На основе описания требований к программной системе разработать его проект в рамках объектно-ориентированного подхода, выполнив декомпозицию задачи на отдельные классы.
6.3 Шаблоны проектирования	Определение шаблона проектирования. Способ описания шаблонов проектирования. Принципы работы с шаблонами проектирования. Типы шаблонов проектирования, их назначение и различия.	Реализовать описанную задачу используя шаблон «Абстрактная фабрика» (Abstract factory). Реализовать описанную задачу используя шаблон «Строитель» (Builder). Реализовать описанную задачу используя шаблон «Фабричный метод» (Factory method).

		<p>Реализовать описанную задачу используя шаблон «Отложенная инициализация» (Lazy initialization).</p> <p>Реализовать описанную задачу используя шаблон «Прототип» (Prototype).</p> <p>Реализовать описанную задачу используя шаблон «Одиночка» (Singleton).</p> <p>Реализовать описанную задачу используя шаблон «Адаптер» (Adapter / Wrapper).</p> <p>Реализовать описанную задачу используя шаблон «Мост» (Bridge).</p> <p>Реализовать описанную задачу используя шаблон «Компоновщик» (Composite).</p> <p>Реализовать описанную задачу используя шаблон «Декоратор» (Decorator).</p> <p>Реализовать описанную задачу используя шаблон «Фасад» (Facade).</p> <p>Реализовать описанную задачу используя шаблон «Приспособленец» (Flyweight).</p> <p>Реализовать описанную задачу используя шаблон «Заместитель» (Proxy).</p> <p>Реализовать описанную задачу используя шаблон «Цепочка обязанностей» (Chain of responsibility).</p> <p>Реализовать описанную задачу используя шаблон «Команда» (Command).</p> <p>Реализовать описанную задачу используя шаблон «Интерпретатор» (Interpreter).</p> <p>Реализовать описанную задачу используя шаблон «Итератор» (Iterator).</p> <p>Реализовать описанную задачу используя шаблон «Посредник» (Mediator).</p> <p>Реализовать описанную задачу используя шаблон «Хранитель» (Memento).</p> <p>Реализовать описанную задачу используя шаблон «Наблюдатель» (Observer).</p> <p>Реализовать описанную задачу используя шаблон «Состояние» (State).</p> <p>Реализовать описанную задачу используя шаблон «Стратегия» (Strategy).</p> <p>Реализовать описанную задачу используя шаблон «Шаблонный метод» (Template method).</p> <p>Реализовать описанную задачу используя шаблон «Посетитель» (Visitor).</p>
6.4 Система управления версиями	<p>Понятие системы управления версиями. Их назначение.</p> <p>Основные возможности систем контроля версий.</p> <p>Ежедневный цикл работы с системой управления версиями.</p>	<p>Составьте сценарий взаимодействия с системой контроля версий для осуществления указанной задачи с репозиторием.</p>
<b>7. Тестирование программного обеспечения</b>		
7.1 Основы тестирования. Виды тестирования	<p>Структура процесса разработки ПП с точки зрения тестирования.</p> <p>«Дымовое» тестирование: назначение и способ проведения.</p> <p>Позитивное тестирование: назначение и способ проведения</p> <p>Негативное тестирование : назначение и способ проведения.</p> <p>Функциональное, нагрузочное,</p>	<p>Составить модульный тест для проверки реализации указанной функциональности.</p>

	<p>стресс-тестирование и тестирование стабильности: назначение и способ проведения.</p> <p>Тестирования безопасности: : назначение и способ проведения.</p> <p>Достоинства и недостатки автоматизированного тестирования.</p> <p>Модульное тестирование, его достоинства и недостатки.</p> <p>Системное тестирование: назначение и способ проведения.</p> <p>Регрессионное тестирование: назначение и способ проведения</p> <p>«Тест-кейс»: назначение и структура.</p>	
7.2 Работа с ошибками	<p>Обработка результатов выполнения тестирования безопасности.</p> <p>Обработка результатов выполнения модульного тестирования.</p> <p>Обработка результатов выполнения тест-кейсов.</p>	
<b>8. Сопровождение программных систем</b>		
8.1 Организация и управление процессом сопровождения. Ресурсы, необходимые для сопровождения	<p>Сущность процесса сопровождения ПО.</p> <p>Основные стандарты, применяемые при организации сопровождения.</p> <p>Виды работ, выполняемые при сопровождении ПО.</p> <p>Отличия реинжиниринга от обычного процесса разработки.</p> <p>Как можно оценить трудозатраты на сопровождение?</p>	Для указанного ПО оценить затраты на его сопровождение.
<b>9. Качество программного обеспечения</b>		
9.1 Основы качества программного обеспечения. Метрики и атрибуты качества	<p>Состав понятия качества ПО.</p> <p>Основные цели и задачи системы управления качеством.</p> <p>Стандарты в области качества ПО.</p> <p>В чем суть инженерии качества?</p> <p>Каковы основные аспекты и уровни модели качества ПО?</p> <p>Определите характеристики качества ПО и их назначение.</p>	
9.2 Управление качеством. Надежность как главная составляющая качества	<p>Методы определения показателей качества.</p> <p>Метрики программного продукта и их составляющие.</p> <p>Модели надежности ПО.</p> <p>Методы оценивания надежности ПО.</p>	По приведенным данным оценить надежность ПО.