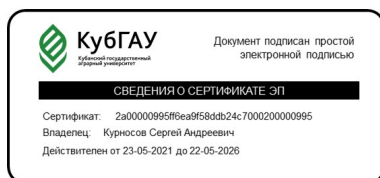


МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
имени И.Т. ТРУБИЛИНА»

Факультет прикладной информатики
Системного анализа и обработки информации



УТВЕРЖДЕНО:
Декан, Руководитель подразделения
Курносое С.А.
(протокол от 22.04.2024 № 7)

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)
« ПРОГРАММНАЯ ИНЖЕНЕРИЯ»**

Уровень высшего образования: бакалавриат

Направление подготовки: 09.03.03 Прикладная информатика

Направленность (профиль): Менеджмент проектов в области информационных технологий,
создание и поддержка информационных систем

Квалификация (степень) выпускника: Бакалавр

Формы обучения: очная, заочная

Год набора: 2024

Срок получения образования: Очная форма обучения – 4 года
Заочная форма обучения – 4 года 8 месяца(-ев)

Объем: в зачетных единицах: 4 з.е.
в академических часах: 144 ак.ч.

2024

Разработчики:

Доцент, кафедра системного анализа и обработки информации Ефанова Н.В.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по направлению подготовки Направление подготовки: 09.03.03 Прикладная информатика, утвержденного приказом Минобрнауки России от 19.09.2017 №922, с учетом трудовых функций профессиональных стандартов: "Специалист по информационным системам", утвержден приказом Минтруда России от 13.07.2023 № 586н; "Руководитель проектов в области информационных технологий", утвержден приказом Минтруда России от 27.04.2023 № 369н.

Согласование и утверждение

№	Подразделение или коллегиальный орган	Ответственное лицо	ФИО	Виза	Дата, протокол (при наличии)
1	Системного анализа и обработки информации	Заведующий кафедрой, руководитель подразделения, реализующего ОП	Барановская Т.П.	Согласовано	08.04.2024, № 8
2	Факультет прикладной информатики	Председатель методической комиссии/совета	Крамаренко Т.А.	Согласовано	22.04.2024, № 8
3	Информационных систем	Руководитель образовательной программы	Замотайлова Д.А.	Согласовано	22.04.2024, № 7

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - формирование у обучающихся теоретических и практических навыков по изучению и использованию современных технологий профессиональной разработки программного обеспечения

Задачи изучения дисциплины:

- научиться использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;
- участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью ;
- разрабатывать и адаптировать алгоритмы и прикладное программное обеспечение, пригодные для практического применения ;
- проводить тестирование компонентов программного обеспечения ИС;
- получить знания о принципах управления проектами создания информационных систем на стадиях жизненного цикла .

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Компетенции, индикаторы и результаты обучения

ОПК-2 Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности

ОПК-2.1 Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

Знать:

ОПК-2.1/Зн1 Современные информационные технологии для решения задач профессиональной деятельности

ОПК-2.1/Зн2 Современные программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

ОПК-2.2 Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

Уметь:

ОПК-2.2/Ум1 Выбирать современные информационные технологии для решения задач профессиональной деятельности

ОПК-2.2/Ум2 Выбирать современные программные средства, в том числе отечественного производства для решения задач профессиональной деятельности

ОПК-2.3 Владеет навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности

Владеть:

ОПК-2.3/Нв1 Навыками применения современных информационных технологий для решения задач профессиональной деятельности

ОПК-2.3/Нв2 Навыками применения современных программных средств, в том числе отечественного производства для решения задач профессиональной деятельности

ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью

ОПК-4.1 Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы

Знать:

ОПК-4.1/Зн1 Стандарты оформления технической документации

ОПК-4.2 Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы

Уметь:

ОПК-4.2/Ум1 Применять стандарты оформления технической документации

ОПК-4.3 Владеет навыками составления технической документации на различных этапах жизненного цикла информационной системы

Владеть:

ОПК-4.3/Нв1 Навыками составления технической документации

ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения

ОПК-7.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий

Знать:

ОПК-7.1/Зн1 Основные языки программирования

ОПК-7.1/Зн3 Современные программные среды разработки информационных систем и технологий

ОПК-7.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ

Уметь:

ОПК-7.2/Ум1 Применять языки программирования

ОПК-7.2/Ум3 Применять современные программные среды разработки информационных систем и технологий

ОПК-7.3 Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

Владеть:

ОПК-7.3/Нв1 Навыками программирования прототипов программно-технических комплексов задач

ОПК-7.3/Нв2 Навыками отладки прототипов программно-технических комплексов задач

ОПК-7.3/Нв3 Навыками тестирования прототипов программно-технических комплексов задач

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

ОПК-8.1 Знает основные технологии создания и внедрения информационных систем, стандарты управления жизненным циклом информационной системы

Знать:

ОПК-8.1/Зн1 Основные технологии создания информационных систем

ОПК-8.1/Зн2 Основные технологии внедрения информационных систем

ОПК-8.1/Зн3 Стандарты управления жизненным циклом информационной системы

ОПК-8.2 Умеет осуществлять организационное обеспечение выполнения работ на всех стадиях и в процессах жизненного цикла информационной системы

Уметь:

ОПК-8.2/Ум1 Осуществлять организационное обеспечение выполнения работ на всех стадиях жизненного цикла информационной системы

ОПК-8.3 Владеет навыками составления плановой и отчетной документации по управлению проектами создания информационных систем на стадиях жизненного цикла

Владеть:

ОПК-8.3/Нв1 Навыками составления плановой документации по управлению проектами создания информационных систем

ОПК-8.3/Нв2 Навыками составления отчетной документации по управлению проектами создания информационных систем

ПК-П2 Способен разрабатывать и адаптировать прикладное программное обеспечение

ПК-П2.1 Знает теоретическое и практическое содержание этапов процесса внедрения, адаптации и настройки прикладного программного обеспечения

Знать:

ПК-П2.1/Зн1 Основы управления изменениями

ПК-П2.1/Зн2 Инструменты и методы анализа требований

ПК-П2.1/Зн3 Возможности ИС

ПК-П2.1/Зн4 Инструменты и методы коммуникаций

ПК-П2.1/Зн5 Каналы коммуникаций

ПК-П2.1/Зн6 Модели коммуникаций

Уметь:

ПК-П2.1/Ум1 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий)

ПК-П2.1/Ум2 Работать в системе учета требований проекта в области ИТ

ПК-П2.1/Ум3 Осуществлять коммуникации в проекте в области ИТ

Владеть:

ПК-П2.1/Нв1 Контроль фактического внесения изменений в элементы ИС

ПК-П2.1/Нв2 Изменение статуса проверенных запросов на изменение в системе учета требований проекта в области ИТ

ПК-П2.2 Умеет организовывать и управлять процессом внедрения, адаптации и настройки прикладного программного обеспечения, в т.ч., распределять работы, выделять ресурсы, контролировать исполнение

Знать:

ПК-П2.2/Зн1 Языки программирования и работы с базами данных

ПК-П2.2/Зн2 Инструменты и методы модульного тестирования

ПК-П2.2/Зн4 Инструменты и методы прототипирования пользовательского интерфейса
ПК-П2.2/Зн5 Возможности типовой ИС
ПК-П2.2/Зн6 Предметная область автоматизации
ПК-П2.2/Зн7 Технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии
ПК-П2.2/Зн14 Основы программирования
ПК-П2.2/Зн15 Современные объектно-ориентированные языки программирования
ПК-П2.2/Зн16 Современные структурные языки программирования
ПК-П2.2/Зн17 Языки современных бизнес-приложений
ПК-П2.2/Зн23 Отраслевая нормативная техническая документация
ПК-П2.2/Зн24 Источники информации, необходимой для профессиональной деятельности
ПК-П2.2/Зн25 Лучшие практики создания (модификации) и сопровождения ИС в экономике

Уметь:

ПК-П2.2/Ум1 Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ИС
ПК-П2.2/Ум2 Тестировать результаты прототипирования ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС
ПК-П2.2/Ум3 Проводить презентации в рамках выполнения работ по созданию (модификации) и сопровождению ИС
ПК-П2.2/Ум4 Проводить переговоры в рамках выполнения работ по созданию (модификации) и сопровождению ИС
ПК-П2.2/Ум5 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) в рамках выполнения работ по созданию (модификации) и сопровождению ИС

Владеть:

ПК-П2.2/Нв1 Разработка прототипа ИС на базе типовой ИС в соответствии с требованиями заказчика к ИС
ПК-П2.2/Нв2 Тестирование прототипа ИС на корректность архитектурных решений
ПК-П2.2/Нв3 Обработка результатов тестирования прототипа ИС на корректность архитектурных решений
ПК-П2.2/Нв5 Согласование пользовательского интерфейса ИС с заказчиком ИС

ПК-П2.3 Владеет навыками внедрения, адаптации и настройки прикладного программного обеспечения, обеспечивая соответствие и контроль разработанного кода и процесса кодирования принятым в организации регламентам и стандартам

Знать:

ПК-П2.3/Зн1 Инструменты и методы верификации структуры программного кода
ПК-П2.3/Зн2 Регламенты кодирования на языках программирования
ПК-П2.3/Зн3 Возможности ИС
ПК-П2.3/Зн4 Предметная область автоматизации
ПК-П2.3/Зн5 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ИС
ПК-П2.3/Зн6 Лучшие практики создания (модификации) и сопровождения ИС в экономике

ПК-П2.3/Зн9 Методы управления содержанием проекта: документирование требований, анализ продукта, модерируемые совещания

Уметь:

ПК-П2.3/Ум1 Распределять работы и выделять ресурсы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П2.3/Ум2 Контролировать исполнение поручений в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

Владеть:

ПК-П2.3/Нв1 Обеспечение соответствия разработанного кода ИС и процесса создания программного кода ИС принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П2.3/Нв2 Назначение и распределение ресурсов в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П2.3/Нв3 Контроль соответствия разработанного кода ИС и процесса создания программного кода ИС принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8 Способен проводить тестирование компонентов программного обеспечения ИС

ПК-П8.1 Знает основные методы тестирования информационных систем и их компонентов

Знать:

ПК-П8.1/Зн1 Инструменты и методы модульного тестирования

ПК-П8.1/Зн2 Предметная область автоматизации

ПК-П8.1/Зн5 Устройство и функционирование современных ИС

ПК-П8.1/Зн10 Современные методики тестирования разрабатываемых ИС: инструменты и методы модульного тестирования

ПК-П8.1/Зн11 Источники информации, необходимой для профессиональной деятельности при выполнении работ по созданию (модификации) и сопровождению ИС

ПК-П8.1/Зн12 Лучшие практики создания (модификации) и сопровождения ИС в экономике

Уметь:

ПК-П8.1/Ум1 Проверять на корректность отдельные модули кода ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС

ПК-П8.1/Ум2 Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ИС

ПК-П8.1/Ум3 Тестировать результаты кодирования ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС

Владеть:

ПК-П8.1/Нв1 Тестирование разрабатываемого модуля ИС

ПК-П8.1/Нв2 Устранение обнаруженных несоответствий в модуле ИС

ПК-П8.2 Умеет тестировать информационные системы и их компоненты различными способами

Знать:

ПК-П8.2/Зн1 Инструменты и методы модульного тестирования

ПК-П8.2/Зн2 Регламенты модульного тестирования

ПК-П8.2/Зн3 Возможности ИС

ПК-П8.2/Зн4 Предметная область автоматизации

ПК-П8.2/Зн5 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ИС

ПК-П8.2/Зн6 Лучшие практики создания (модификации) и сопровождения ИС в экономике

Уметь:

ПК-П8.2/Ум1 Распределять работы и выделять ресурсы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.2/Ум2 Выдавать поручения и контролировать их исполнение в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

Владеть:

ПК-П8.2/Нв1 Обеспечение соответствия процессов модульного тестирования ИС принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.2/Нв2 Назначение и распределение ресурсов в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.2/Нв3 Контроль исполнения в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.3 Владеет навыками тестирования информационных систем и компонентов программного обеспечения информационных систем

Знать:

ПК-П8.3/Зн1 Инструменты и методы проектирования архитектуры и дизайна ИС

ПК-П8.3/Зн2 Инструменты и методы верификации архитектуры и дизайна ИС

ПК-П8.3/Зн3 Возможности ИС

ПК-П8.3/Зн4 Предметная область автоматизации

ПК-П8.3/Зн8 Основы программирования

ПК-П8.3/Зн9 Современные объектно-ориентированные языки программирования

ПК-П8.3/Зн10 Современные структурные языки программирования

ПК-П8.3/Зн11 Языки современных бизнес-приложений

ПК-П8.3/Зн12 Современные методики тестирования разрабатываемых ИС: инструменты и методы модульного тестирования, инструменты и методы тестирования нефункциональных и функциональных характеристик ИС

ПК-П8.3/Зн13 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ИС

ПК-П8.3/Зн14 Лучшие практики создания (модификации) и сопровождения ИС в экономике

Уметь:

ПК-П8.3/Ум1 Проектировать архитектуру и дизайн ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.3/Ум2 Проверять (верифицировать) архитектуру и дизайн ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.3/Ум3 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

Владеть:

ПК-П8.3/Нв1 Воспроизведение зафиксированных в системе учета дефектов и несоответствий в архитектуре и дизайне ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

ПК-П8.3/Нв2 Установление причин возникновения дефектов и несоответствий в архитектуре и дизайне ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

3. Место дисциплины в структуре ОП

Дисциплина (модуль) «Программная инженерия» относится к обязательной части образовательной программы и изучается в семестре(ах): Очная форма обучения - 5, Заочная форма обучения - 5.

В процессе изучения дисциплины студент готовится к видам профессиональной деятельности и решению профессиональных задач, предусмотренных ФГОС ВО и образовательной программой.

4. Объем дисциплины и виды учебной работы

Очная форма обучения

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Пятый семестр	144	4	65	3	30	32	52	Экзамен (27)
Всего	144	4	65	3	30	32	52	27

Заочная форма обучения

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Пятый семестр	144	4	17	3	10	4	127	Контроль ная работа Экзамен
Всего	144	4	17	3	10	4	127	

5. Содержание дисциплины

5.1. Разделы, темы дисциплины и виды занятий
(часы промежуточной аттестации не указываются)

Очная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие результатам освоения программы
Раздел 1. Введение в дисциплину программной инженерии	8			4	4	ОПК-2.1 ОПК-2.2 ОПК-2.3
Тема 1.1. Введение в дисциплину профессиональной разработки программного обеспечения	4			2	2	
Тема 1.2. Жизненный цикл ИТ-проекта и продукта	4			2	2	
Раздел 2. Стандартизация деятельности в области программной инженерии	12		4	2	6	ОПК-4.1 ОПК-4.2 ОПК-4.3
Тема 2.1. Стандарты и нормативная документация	12		4	2	6	
Раздел 3. Методологические основы разработки программных продуктов	4			2	2	ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 3.1. Методологии и технологии разработки программных продуктов	4			2	2	
Раздел 4. Основы управления ИТ-проектами	6		2	2	2	ОПК-8.1 ОПК-8.2 ОПК-8.3
Тема 4.1. Основные принципы и стандарты управления ИТ-проектами	6		2	2	2	
Раздел 5. Процессы профессиональной разработки программных продуктов	70		20	18	32	ПК-П2.1 ПК-П2.2 ПК-П2.3
Тема 5.1. Управление требованиями	18		4	4	10	
Тема 5.2. Проектирование и разработка программного обеспечения	44		14	10	20	
Тема 5.3. Профессиональные тренды в разработке программного обеспечения	8		2	4	2	
Раздел 6. Процессы тестирования программных продуктов	14		4	4	6	ПК-П8.1 ПК-П8.2 ПК-П8.3

Тема 6.1. Тестирование программных продуктов	14		4	4	6	
Раздел 7. Промежуточная аттестация	3	3				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-4.1 ОПК-4.2 ОПК-4.3 ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 7.1. Экзамен	3	3				ОПК-8.1 ОПК-8.2 ОПК-8.3 ПК-П2.1 ПК-П2.2 ПК-П2.3 ПК-П8.1 ПК-П8.2 ПК-П8.3
Итого	117	3	30	32	52	

Заочная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие результатам освоения программы
Раздел 1. Введение в дисциплину программной инженерии	22			2	20	ОПК-2.1 ОПК-2.2 ОПК-2.3
Тема 1.1. Введение в дисциплину профессиональной разработки программного обеспечения	11			1	10	
Тема 1.2. Жизненный цикл ИТ-проекта и продукта	11			1	10	
Раздел 2. Стандартизация деятельности в области программной инженерии	13		2	1	10	ОПК-4.1 ОПК-4.2 ОПК-4.3
Тема 2.1. Стандарты и нормативная документация	13		2	1	10	
Раздел 3. Методологические основы разработки программных продуктов	11			1	10	ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 3.1. Методологии и технологии разработки программных продуктов	11			1	10	

Раздел 4. Основы управления ИТ-проектами	10				10	ОПК-8.1 ОПК-8.2 ОПК-8.3
Тема 4.1. Основные принципы и стандарты управления ИТ-проектами	10				10	
Раздел 5. Процессы профессиональной разработки программных продуктов	63		6		57	ПК-П2.1 ПК-П2.2 ПК-П2.3
Тема 5.1. Управление требованиями	12		2		10	
Тема 5.2. Проектирование и разработка программного обеспечения	41		4		37	
Тема 5.3. Профессиональные тренды в разработке программного обеспечения	10				10	
Раздел 6. Процессы тестирования программных продуктов	22		2		20	ПК-П8.1 ПК-П8.2 ПК-П8.3
Тема 6.1. Тестирование программных продуктов	22		2		20	
Раздел 7. Промежуточная аттестация	3	3				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-4.1 ОПК-4.2 ОПК-4.3 ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 7.1. Экзамен	3	3				ОПК-8.1 ОПК-8.2 ОПК-8.3 ПК-П2.1 ПК-П2.2 ПК-П2.3 ПК-П8.1 ПК-П8.2 ПК-П8.3
Итого	144	3	10	4	127	

5. Содержание разделов, тем дисциплин

Раздел 1. Введение в дисциплину программной инженерии

(Заочная: Лекционные занятия - 2ч.; Самостоятельная работа - 20ч.; Очная: Лекционные занятия - 4ч.; Самостоятельная работа - 4ч.)

Тема 1.1. Введение в дисциплину профессиональной разработки программного обеспечения

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

1. Инженерное дело и программная инженерия
2. Типы программных продуктов
3. Защита программного обеспечения
4. Понятие ИТ-проекта
5. Инструментальные средства разработки

Тема 1.2. Жизненный цикл ИТ-проекта и продукта

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

1. Понятие ЖЦ.
2. Модели ЖЦ.

Раздел 2. Стандартизация деятельности в области программной инженерии

(Заочная: Лабораторные занятия - 2ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 6ч.)

Тема 2.1. Стандарты и нормативная документация

(Заочная: Лабораторные занятия - 2ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 6ч.)

1. Основные стандарты программной инженерии
2. Техническое задание
3. Спецификация требований

Раздел 3. Методологические основы разработки программных продуктов

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

Тема 3.1. Методологии и технологии разработки программных продуктов

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

1. Предиктивный и гибкий подход
2. Сравнительный анализ методологий
3. Технологии разработки ПО

Раздел 4. Основы управления ИТ-проектами

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

Тема 4.1. Основные принципы и стандарты управления ИТ-проектами

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

1. Проекты разработки ПО
2. Лучшие практики разработки ПО.
3. Гибридный подход

Раздел 5. Процессы профессиональной разработки программных продуктов

(Заочная: Лабораторные занятия - 6ч.; Самостоятельная работа - 57ч.; Очная: Лабораторные занятия - 20ч.; Лекционные занятия - 18ч.; Самостоятельная работа - 32ч.)

Тема 5.1. Управление требованиями

(Заочная: Лабораторные занятия - 2ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 10ч.)

Принятая терминология. Классификация требований. Процесс сбора и анализа требований. Методы представления требований

Тема 5.2. Проектирование и разработка программного обеспечения

(Заочная: Лабораторные занятия - 4ч.; Самостоятельная работа - 37ч.; Очная: Лабораторные занятия - 14ч.; Лекционные занятия - 10ч.; Самостоятельная работа - 20ч.)

1. Объектно-ориентированный анализ и проектирование (методология RUP UML) (6 ч)
2. Основы дизайна интерфейса пользователя. (4 ч)

Тема 5.3. Профессиональные тренды в разработке программного обеспечения

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

1. Настольные и сетевые приложения
2. Программирование робототехники
3. Специфика «-техов» (edtech, med-, fin-, food-, agro-)
4. Машинное обучение и анализ данных

Раздел 6. Процессы тестирования программных продуктов

(Заочная: Лабораторные занятия - 2ч.; Самостоятельная работа - 20ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 6ч.)

Тема 6.1. Тестирование программных продуктов

(Заочная: Лабораторные занятия - 2ч.; Самостоятельная работа - 20ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 6ч.)

1. Терминология и основные положения.
2. Виды и техники тестирования (классификация).
3. План тестирования, тестовое покрытие, тестовые процедуры, протоколы и отчеты тестирования ИС.

Раздел 7. Промежуточная аттестация

(Заочная: Внеаудиторная контактная работа - 3ч.; Очная: Внеаудиторная контактная работа - 3ч.)

Тема 7.1. Экзамен

(Заочная: Внеаудиторная контактная работа - 3ч.; Очная: Внеаудиторная контактная работа - 3ч.)

Проведение промежуточной аттестации в форме экзамена

6. Оценочные материалы текущего контроля

Раздел 1. Введение в дисциплину программной инженерии

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Преимущества использования ООП:
набор новых средств программирования, добавленных в язык
использование готовых методов, процедур, функций для решения разных задач
существенно облегчает программирование, даже при незнании структуры языка
используется интерактивная модель программирования
2. Язык программирования является объектно-ориентированным, если

Он поддерживает абстрактные типы данных, которыми являются объекты с определенными интерфейсами (средствами взаимодействия с внешней средой) и скрытым внутренним состоянием

Он имеет возможность визуализации и интерактивности программы

Объекты имеют связанные с ними типы (классы)

Поддерживаются механизмы наследования классов

3. Основные свойства ООП:

инкапсуляция

легкость и простота использования

наследование

полиморфизм

интерактивность

4. Объект состоит из следующих трех частей:

имя объекта

состояние (переменные состояния)

методы (операции)

спецификаторов доступа

условных обозначений

5. ... - это механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использования

Примечание. Ответ записать в нижнем регистре в именительном падеже

6. ... - это процесс, посредством которого, один объект может наследовать свойства другого объекта и добавлять к ним черты, характерные только для него

Примечание. Ответ записать в нижнем регистре в именительном падеже

7. Установите правильную последовательность стадий жизненного цикла проекта

Инициация

Планирование

Реализация

Завершение

8. Одна итерация в Agile-фреймворке SCRUM называется ...

(записать на русском языке в именительном падеже)

Раздел 2. Стандартизация деятельности в области программной инженерии

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Нормативный документ, который устанавливает комплекс норм, правил и требований к объекту стандартизации – это ...

(ответ записать в нижнем регистре в именительном падеже)

2. Деятельность, направленная на достижение упорядочения в определенной области посредством установления положений для всеобщего и многократного применения в отношении реально существующих и потенциальных задач - это ...

(ответ записать в нижнем регистре в именительном падеже)

3. Объектом стандартизации является:

Продукция

Процесс

Услуга

Информация

Данные

4. Согласованный набор стандартов и/или их фрагментов, созданный для нормативного обеспечения конкретного проекта или задачи - это ... стандартов

(ответ записать в нижнем регистре в именительном падеже)

5. Совокупность действий и процедур с целью подтверждения того, что товар соответствует требованиям стандартов - это ...

(ответ записать в нижнем регистре в именительном падеже)

6. Укажите номер ГОСТ (без года принятия), определяющий процессы жизненного цикла программных средств

(в ответе записать только числовое значение)

7. Соотнесите уровень стандартизации с его определением

Уровень стандартизации

1. Международная стандартизация
2. Региональная стандартизация
3. Национальная стандартизация

Определение уровня

А. стандартизация, участие в которой открыто для национальных всех стран мира

Б. стандартизация, участие в которой открыто для национальных органов по стандартизации только одного географического, политического или экономического региона

В. стандартизация, проводимая на уровне одной конкретной страны

8. Принятые и адаптированные международные стандарты классифицируются на:

Идентичные

Измененные

Добавленные

Аналогичные

Не эквивалентные

9. Соотнесите названия и определения классификаций для принятых и адаптированных международных стандартов

Название категории стандартов

1. Идентичные(IDT)
2. Измененные(MOD)
3. Не эквивалентный(NEQ)

Характеристика категории стандартов

А. Идентичные по техническому содержанию и структуре, но могут содержать минимальные редакционные изменения

Б. Принятые стандарты содержат технические отклонения, которые ясно идентифицированы и объяснены

В. Региональный или национальный стандарт не эквивалентен международным стандартам, изменения ясно не идентифицированы, и не установлено четкое соответствие

10. Установите правильную последовательность этапов разработки стандарта в целях обеспечения организационно-методического единства и создания условий для своевременной подготовки к применению стандартов

Организация разработки стандарта;

Разработка проекта стандарта (первой редакции);

Разработка проекта стандарта (окончательной редакции) и представление его на утверждение;

Утверждение и государственная регистрация стандарта;

Издание стандарта;

11. Нормы и правила, регламентирующие рабочую деятельность всех сотрудников компании, алгоритм взаимодействия с потребителем товаров или услуг, предлагаемых организацией:

Организационные стандарты

Корпоративные стандарты

Коммуникативные стандарты

Комплексные стандарты

12. Расположите стадии создания системы, предусмотренные в стандарте ISO/IEC 15288 в хронологическом порядке

Формирование концепции
Разработка
Реализация
Эксплуатация
Снятие с эксплуатации

13. Какой правовой инструмент определяет использование и распространение программного обеспечения, защищенного авторскими правами?

(ответ записать в нижнем регистре в именительном падеже)

Раздел 3. Методологические основы разработки программных продуктов

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Перегруженные функции различаются компилятором с помощью

Разного кода записанного в теле функций
Сигнатуры функций
Типа возвращаемого функцией значения
Имени функции

2. Выберите вариант правильного задания заголовка шаблона функции в C++

template<class имя_типа>
template<const имя_типа>
template<new имя_типа>
template(имя_типа)

3. К свойствам параметров шаблона относятся:

Имена параметров шаблона могут повторно использоваться во всем определении шаблона
Список параметров шаблона функций не может быть пустым
В заголовке шаблона можно использовать параметры с одинаковыми именами
Имя параметра шаблона имеет в определяемой шаблоном функции все права имени типа

4. Для осуществления перегрузки функций необходимо

Перед определением функции указать спецификатор inline
Определить функции с одинаковым именем, но разными сигнатурами
Определить функции с разными именами, но одинаковыми сигнатурами
Перед определением функции указать спецификатор overload

5. Перегруженные функции используются для выполнения

Одинаковых операций над различными типами данных
Разных операций над одинаковыми типами данных
Сходных операций над различными типами данных
Разных операций над различными типами данных

6. Целью введения шаблонов функции является

Автоматическая подстановка в место вызова функции её кода
Автоматизация создания функций, которые могут обрабатывать разнотипные данные
Автоматическая замена функции в тексте программы на одноименную функцию, указанную в шаблоне функций
Замена любой функция в программе на ту, что задана шаблоном

7. Параметры шаблона НЕ обладают следующими свойствами:

Имена параметров шаблона могут повторно использоваться во всем определении шаблона
Список параметров шаблона функций не может быть пустым
В заголовке шаблона можно использовать параметры с одинаковыми именами
Имя параметра шаблона имеет в определяемой шаблоном функции все права имени типа

8. Механизм шаблонов функций позволяет

Автоматизировать подготовку определений перегруженных функций

Автоматизировать подстановку в место вызова функции её кода

Автоматизировать замену функций в тексте программы на одноименную функцию, указанную в шаблоне функций

Любую функцию в программе автоматически заменить той, что указана в шаблоне функций

9. Простейшим образом класс можно определить с помощью конструкции

имя_класса ключ_класса {список компонентов}

ключ_класса имя_класса {список компонентов}

ключ_класса {список компонентов}

имя_класса {список компонентов}

10. В качестве ключа класса в C++ может использоваться ключевое слово

class

struct

union

main

enum

key

11. Для инициализации объектов класса используется ... класса.

(ответ записать в нижнем регистре в именительном падеже)

12. Для уничтожения динамически размещенного объекта класса и освобождении

занимаемой им памяти используется ... класса

(ответ записать в нижнем регистре в именительном падеже)

13. К спецификаторам доступа к объектам класса относятся

struct

public

private

protected

class

union

14. Объявление статических данных класса в C++ начинается с ключевого слова

(ответ записать в нижнем регистре)

15. Функция в классе в C++, которая не являясь его компонентом, имеет доступ к его

защищенным и собственным компонентам, называется ... (какой?)

(ответ записать в нижнем регистре)

16. Объявление дружественной функции начинается с ключевого слова

(ответ записать в нижнем регистре)

17. Дружественная функция

Не может быть компонентной функцией того класса, по отношению к которому определяется как дружественная

Всегда является компонентом класса

Может быть компонентной функцией другого, ранее определенного класса

Может быть глобальной функцией

Раздел 4. Основы управления ИТ-проектами

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Проектная деятельность отличается от процессной деятельности тем, что ...

Проект является непрерывной деятельностью, а процесс – единоразовым мероприятием

Проект поддерживает неизменность организации, а процессы способствуют ее изменению

Процессы в организации цикличны, они повторяются, а проект – уникален, он всегда имеет дату начала и окончания

Процессы в организации регламентируются документально, проекты не требуют документального оформления

2. ... проекта - это среда проекта, порождающая совокупность внутренних или внешних сил, которые способствуют или мешают достижению цели проекта

Примечание. Ответ записать в нижнем регистре в именительном падеже

3. ... - это ограниченное по времени, целенаправленное изменение отдельной системы с установленными требованиями к качеству результатов, с ограничениями расходования средств

Примечание. Ответ дать в нижнем регистре в именительном падеже

4. Горизонтальная линейная диаграмма, на которой задачи проекта представляются протяженными во времени отрезками, характеризующимися датами начала и окончания, задержками и, возможно, другими временными параметрами, — это диаграмма ...

(в ответ записать название)

5. Графическое отображение работ проекта и зависимостей между ними представляет собой КАКУЮ диаграмму?

(ответ записать в нижнем регистре)

6. Основной структурной единицей участников проекта является ... проекта

Команда

Отдел

Ячейка

Группа

7. Завершение проекта — это стадия процесса управления проектом, включающая процессы ...

Формирования концепции проекта

Формирования сводного плана проекта

Осуществления всех запланированных проектных работ

Ввода в эксплуатацию и принятия проекта заказчиком, документирования и анализа опыта реализации проекта

8. По результатам постпроектной оценки всегда составляется детальный ...

(ответ записать в нижнем регистре в именительном падеже)

9. Желаемый результат деятельности, достигаемый в итоге успешного осуществления проекта в заданных условиях его выполнения - это ... проекта

(ответ записать в нижнем регистре в именительном падеже)

10. Непосредственное инициирование проекта включает в себя ...

Принятие решения о начале проекта

Определение и назначение управляющего проектом

Принятие решения об обеспечении ресурсами выполнения первой фазы проекта

Анализ проблемы и потребности в проекте

Сбор исходных данных

Организация и контроль выполнения работ

Утверждение окончательного сводного плана управления проектом

11. Правильная последовательность процессов определения целей и задач

Формулирование

Структурирование

Согласование

Фиксация

12. ... проекта - это документ, разработанный вышестоящей администрацией, который предоставляет менеджеру проекта право использовать ресурсы организации для выполнения работ.

Примечание. Ответ записать в нижнем регистре в именительном падеже

13. Расположите этапы создания проектной команды в верном порядке:

Формирование

"Притирка" участников

Нормальное функционирование

Продуктивная работа

Раздел 5. Процессы профессиональной разработки программных продуктов

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. На начальном этапе разработки программного продукта предпринимаются следующие действия:

написание кода

тестирование

выявление требований и ограничений

определение эффективности разработки

2. Стадия конструирования имеет следующее назначение:

Создание программного продукта с начальной функциональностью

Создать точную иерархическую модель проекта

Создать архитектурный базис

Создать архитектурный каркас

3. Руководство пользователя производится на:

стадии уточнения

стадии конструирования

начальной стадии

4. Конкретное амплуа сотрудника в конкретном проекте в определенное время называется ... (в именительном падеже)

(ответ записать в нижнем регистре в именительном падеже)

Раздел 6. Процессы тестирования программных продуктов

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Какой член команды проекта (его роль в команде) несет ответственность за качество программного обеспечения (функциональность, надежность, эффективность и т.п.):

Примечание. В ответе записать название роли в именительном падеже в нижнем регистре.

2. Обнаруженные в процессе тестирования или наблюдения программная ошибка, а также несоответствие спецификации называется

Программный дефект

Уязвимость

Неточность

Нет верного ответа

3. При тестировании каким(и) методом(методами) разработчик имеет доступ к исходному программному коду и может писать код, который связан с библиотеками тестируемого ПО?

Серым ящиком

Белым ящиком

Черным ящиком

4. При тестировании каким методом, тестирующий имеет доступ к ПО только через те же интерфейсы, что и заказчик или пользователь, либо через внешние интерфейсы, позволяющие другому компьютеру либо другому процессу подключиться к системе для тестирования

Серым ящиком

Черным ящиком

Белым ящиком

5. Обнаружение, локализация и устранение ошибок в программе - это ... программы.

Примечание. Ответ записать в нижнем регистре в именительном падеже

6. ... - это проверка соответствия программы требованиям, осуществляемая путем наблюдения за ее работой в специальных, искусственно созданных ситуациях, выбранных определенным образом.

Примечание. Ответ записать в нижнем регистре в именительном падеже

7. Главной целью тестирования ИС является

- Нахождение ошибок
- Управление изменениями
- Локализация неисправностей
- Реализация требований к характеристикам программы

8. Что из перечисленного ниже пунктов является объектом тестирования?

- Работа программы
- Качество кода
- Возможность программы работы на других платформах
- Продаваемость
- Степень соответствия
- Объективность кода

9. Что из перечисленного НЕ входит в фазу исправления идентифицированных ошибок?

- Локализация ошибок
- Нахождение причин ошибок
- Корректировка программы
- Выбор языка программирования

10. Виды тестирования информационной системы можно разделить на две большие группы:

- Функциональное
- Компонентное
- Нагрузочное
- Конфигурационное
- Нефункциональное
- Модульное

11. Какое тестирование призвано показать, ИС предоставляет пользователям ровно ту функциональность, которую они от нее ожидают?

Примечание. Ответ записать в нижнем регистре

12. Испытание информационной системы на этапах подготовки к эксплуатации включает в себя:

- Тестирование процесса установки
- Регрессионное тестирование
- Приемочное тестирование
- Тестирование на различных конфигурациях
- Предварительное или дымовое тестирование

13. Расположите в правильном порядке разделы плана тестирования

- Введение
- Тестовые требования
- Стратегия тестирования
- Ресурсы
- Описание результатов тестирования

14. Что из перечисленного является одной из метрик оценки качества тестирования, представляющая из себя плотность покрытия тестами требований либо исполняемого кода?

- План тестирования
- Тестовое покрытие
- Тестовые процедуры
- Протоколы и отчеты тестирования ИС.

15. ... - это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.

Примечание. Ответ записать в нижнем регистре

16. Подтверждение соответствия конечного продукта предопределённым требованиям – это ...

Примечание. Ответ записать в нижнем регистре в именительном падеже

17. Тестирования, которое проводится без использования инструментов автоматизации, при этом долго и дорого - это какое тестирование?

(ответ записать в нижнем регистре)

18. Установите в нужном порядке план работы над тест дизайном

Анализ имеющихся проектных артефактов

Написание спецификации по тест дизайну

Проектирование и создание тестовых случаев или тест-кейсов

19. Установите в нужном порядке уровни тестирования программного обеспечения

Модульное тестирование

Интеграционное тестирование

Системное тестирование

Приемочное тестирование

20. Установите соответствия

Артефакт процесса тестирования

1. План тестирования

2. Набор тест-кейсов и тестов

3. Дефекты / Баг Репорты

Назначение артефакта

А. это документ описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения

Б. это последовательность действий, по которой можно проверить соответствует ли тестируемая функция установленным требованиям

В. это документы, описывающие ситуацию или последовательность действий, приведшую к некорректной работе объекта тестирования, с указанием причин и ожидаемого результата

21. Дефект называется (какой?) ..., если ни один элемент системы не работает, никто не может ей пользоваться.

(ответ записать в нижнем регистре)

22. Установите соответствие дефектов по степени влияния на систему

Дефект

1. Блокирующий

2. Критический

3. Высокий

4. Низкий

5. Незначительный / Тривиальный

Степень влияния

А. Ни один элемент системы не работает: никто не может ей пользоваться

Б. Важная часть системы не работает: пользоваться можно, но вероятность сбоя — высокая

В. Система работает, но не так, как нужно. Пользоваться можно, элементы активны, но выполняют не то, что задумано

Г. С системой всё в порядке, но работать неудобно. Например, дефект в дизайне

Д. Баг, который не влияет на работу программы. Например, орфографическая ошибка в тексте встроенной справки.

23. Установите соответствия дефектов по скорости исправления, приоритету

Приоритет

1. Высокий

2. Средний

3. Низкий

Дефект

А. Без правок ИС работать не может

Б. Ошибка не критичная, но без починки ПО не будет работать, как надо

В. Можно исправить, когда разобрались с критическими ошибками

24. Как называется вид тестовой документации, который описывает процесс проверки сервиса?

Тестовые случаи или тест-кейсы

План тестирования

Шаги воспроизведения

Алгоритм прохождения

25. Выберите обязательные поля для описания тест-кейса:

Шаги

Логи

Ожидаемый результат

Версия приложения

Раздел 7. Промежуточная аттестация

Форма контроля/оценочное средство:

Вопросы/Задания:

.

7. Оценочные материалы промежуточной аттестации

Очная форма обучения, Пятый семестр, Экзамен

Контролируемые ИДК: ОПК-2.1 ОПК-4.1 ОПК-7.1 ОПК-8.1 ОПК-2.2 ОПК-4.2 ОПК-7.2 ОПК-8.2 ОПК-2.3 ОПК-4.3 ОПК-7.3 ОПК-8.3 ПК-П2.1 ПК-П8.1 ПК-П2.2 ПК-П8.2 ПК-П2.3 ПК-П8.3

Вопросы/Задания:

1. Программная инженерия как дисциплина, комплексно охватывающая процесс разработки информационных систем

2. Структурное программирование.

3. Процедурно-ориентированное программирование.

4. Объектно-ориентированное программирование.

5. Модульное программирование.

6. Современные парадигмы программирования.

7. Унифицированный язык моделирования UML

8. Методология проектирования Microsoft.

9. Гибкие методологии разработки ПО: Scrum.

10. Гибкие методологии разработки ПО: Kanban.

11. Спецификация требований: состав и структура документа.
12. Техническое задание: назначение, основные разделы документа.
13. Технический проект: назначение, основные разделы документа.
14. Структура ТЭО.
15. Рабочий проект: назначение, основные разделы документа.
16. Программа тестовых испытаний.
17. Документирование требований – основы языка Gherkin.
18. Понятие и структура тестового скрипта.
19. Документация пользователя.
20. Принципы и инструменты документирования кода.
21. Тестирование как процесс обеспечения качества ПС: понятие, цели, виды тестирования.
22. Автоматизация тестирования.
23. Разработка через тестирование (TDD).
24. Понятие и структура тестового скрипта.
25. Тестовое покрытие.
26. Системы контроля версий. Git.
27. Опытнo-промышленная эксплуатация.
28. Стратегии развертывания системы.
29. Интеграционное тестирование системы.
30. Миграция данных.
31. Основные понятия ООП: инкапсуляция, наследование, полиморфизм.
32. Основы ООП: класс как абстрактный тип, конструкторы и деструкторы.
33. Основы ООП: объявление классов, объекты классов, указатель this.
34. Основы ООП: спецификаторы доступа класса.

35. Основы ООП: «друзья» классов (дружественные функции и классы).
36. Основы ООП: перегрузка и шаблоны функций.
37. Основы ООП: наследование классов, шаблоны классов.
38. Основы ООП: перегрузка стандартных операций.
39. Основы ООП: обработка исключительных ситуаций.
40. Унифицированный язык моделирования UML: цели создания и назначение.
41. Методология проектирования UML: диаграммы вариантов использования.
42. Методология проектирования UML: диаграммы состояний.
43. Методология проектирования UML: диаграммы деятельности.
44. Методология проектирования UML: диаграммы классов.
45. Методология проектирования UML: диаграммы последовательности.
46. Методология проектирования UML: диаграммы кооперации.
47. Методология проектирования UML: диаграммы компонентов.
48. Методология проектирования UML: диаграммы развертывания.
49. Методология проектирования Microsoft: схема слоев – назначение, пример.
50. Понятие и модели жизненного цикла программного средства.
51. Основные процессы жизненного цикла программного средства.
52. Гибкие методологии разработки ПО: сравнительный анализ, Scrumban, преимущества и недостатки.
53. Авторская и коллективная разработка.
54. Общинная модель разработки ПО. Основные отличия от коллективной разработки.
55. Коллективная разработка. Модель группы и иерархическая модель.
56. Коллективная разработка: роли и обязанности членов проектной группы. Деструктивное сочетание ролей.
57. Коллективная разработка: роли, деструктивное и созидательное сочетание ролей.

58. Управление требованиями: определение требований, управления требованиями, место и роль управления требованиями в процессе жизненного цикла ПО.

59. Управление требованиями: классификация требований.

60. Управление требованиями: методы сбора требований.

61. Проектирование и разработка интерфейса ПС: понятие, эргономика, типы диалогов.

62. Проектирование и разработка интерфейса ПС: раскадровка, макетирование.

63. Проектирование и разработка интерфейса ПС: принципы построения интерфейсов.

64. Практика работы с требованиями

описать не менее 5 историй пользователя при выявлении требований к программному продукту. Примеры программных продуктов: будильник, секундомер, блокнот, графический редактор, музыкальный проигрыватель, почтовый клиент, электронный учебник, банкомат, кинокасса, заказ такси, оплата картой, пульсометр

65. Практика проектирования UML

построить требуемую диаграмму UML заданной предметной области (согласно программному продукту). Варианты диаграмм UML: ВИ, последовательности, активности, состояний

66. Практика ООП

описать класс с заданными компонентными данными и функциями. Не менее 2х компонентных данных, минимум 1 конструктор, не менее 3 компонентных функций. Класс по программному продукту

67. Практика Agile

спланировать итерации проекта разработки программного продукта с учетом стандартных этапов жизненного цикла программного продукта. Длительность итерации принять равной 1 дню

68. Практика использования языка Gherkin

описать на языке Gherkin пользовательский сценарий авторизации аккаунта для использования программного продукта. Сценарий должен иметь 1 основной и 1 альтернативный поток

Заочная форма обучения, Пятый семестр, Экзамен

Контролируемые ИДК: ОПК-2.1 ОПК-4.1 ОПК-7.1 ОПК-8.1 ОПК-2.2 ОПК-4.2 ОПК-7.2 ОПК-8.2 ОПК-2.3 ОПК-4.3 ОПК-7.3 ОПК-8.3 ПК-П2.1 ПК-П8.1 ПК-П2.2 ПК-П8.2 ПК-П2.3 ПК-П8.3

Вопросы/Задания:

1. Программная инженерия как дисциплина, комплексно охватывающая процесс разработки информационных систем

2. Структурное программирование.

3. Процедурно-ориентированное программирование.

4. Объектно-ориентированное программирование.

5. Модульное программирование.

6. Современные парадигмы программирования.
7. Унифицированный язык моделирования UML
8. Методология проектирования Microsoft.
9. Гибкие методологии разработки ПО: Scrum.
10. Гибкие методологии разработки ПО: Kanban.
11. Спецификация требований: состав и структура документа.
12. Техническое задание: назначение, основные разделы документа.
13. Технический проект: назначение, основные разделы документа.
14. Структура ТЭО.
15. Рабочий проект: назначение, основные разделы документа.
16. Программа тестовых испытаний.
17. Документирование требований – основы языка Gherkin.
18. Понятие и структура тестового скрипта.
19. Документация пользователя.
20. Принципы и инструменты документирования кода.
21. Тестирование как процесс обеспечения качества ПС: понятие, цели, виды тестирования.
22. Автоматизация тестирования.
23. Разработка через тестирование (TDD).
24. Понятие и структура тестового скрипта.
25. Тестовое покрытие.
26. Системы контроля версий. Git.
27. Опытно-промышленная эксплуатация.
28. Стратегии развертывания системы.
29. Интеграционное тестирование системы.

30. Миграция данных.
31. Основные понятия ООП: инкапсуляция, наследование, полиморфизм.
32. Основы ООП: класс как абстрактный тип, конструкторы и деструкторы.
33. Основы ООП: объявление классов, объекты классов, указатель this.
34. Основы ООП: спецификаторы доступа класса.
35. Основы ООП: «друзья» классов (дружественные функции и классы).
36. Основы ООП: перегрузка и шаблоны функций.
37. Основы ООП: наследование классов, шаблоны классов.
38. Основы ООП: перегрузка стандартных операций.
39. Основы ООП: обработка исключительных ситуаций.
40. Унифицированный язык моделирования UML: цели создания и назначение.
41. Методология проектирования UML: диаграммы вариантов использования.
42. Методология проектирования UML: диаграммы состояний.
43. Методология проектирования UML: диаграммы деятельности.
44. Методология проектирования UML: диаграммы классов.
45. Методология проектирования UML: диаграммы последовательности.
46. Методология проектирования UML: диаграммы кооперации.
47. Методология проектирования UML: диаграммы компонентов.
48. Методология проектирования UML: диаграммы развертывания.
49. Методология проектирования Microsoft: схема слоев – назначение, пример.
50. Понятие и модели жизненного цикла программного средства.
51. Основные процессы жизненного цикла программного средства.
52. Гибкие методологии разработки ПО: сравнительный анализ, Scrum, преимущества и недостатки.
53. Авторская и коллективная разработка.

54. Общинная модель разработки ПО. Основные отличия от коллективной разработки.
55. Коллективная разработка. Модель группы и иерархическая модель.
56. Коллективная разработка: роли и обязанности членов проектной группы. Деструктивное сочетание ролей.
57. Коллективная разработка: роли, деструктивное и созидательное сочетание ролей.
58. Управление требованиями: определение требований, управления требованиями, место и роль управления требованиями в процессе жизненного цикла ПО.
59. Управление требованиями: классификация требований.
60. Управление требованиями: методы сбора требований.
61. Проектирование и разработка интерфейса ПС: понятие, эргономика, типы диалогов.
62. Проектирование и разработка интерфейса ПС: раскадровка, макетирование.
63. Проектирование и разработка интерфейса ПС: принципы построения интерфейсов.
64. Практика работы с требованиями
описать не менее 5 историй пользователя при выявлении требований к программному продукту. Примеры программных продуктов: будильник, секундомер, блокнот, графический редактор, музыкальный проигрыватель, почтовый клиент, электронный учебник, банкомат, кинокасса, заказ такси, оплата картой, пульсометр
65. Практика проектирования UML
построить требуемую диаграмму UML заданной предметной области (согласно программному продукту). Варианты диаграмм UML: ВИ, последовательности, активности, состояний
66. Практика ООП
описать класс с заданными компонентными данными и функциями. Не менее 2х компонентных данных, минимум 1 конструктор, не менее 3 компонентных функций. Класс по программному продукту
67. Практика Agile
спланировать итерации проекта разработки программного продукта с учетом стандартных этапов жизненного цикла программного продукта. Длительность итерации принять равной 1 дню
68. Практика использования языка Gherkin
описать на языке Gherkin пользовательский сценарий авторизации аккаунта для использования программного продукта. Сценарий должен иметь 1 основной и 1 альтернативный поток

Заочная форма обучения, Пятый семестр, Контрольная работа

Контролируемые ИДК: ОПК-2.1 ОПК-4.1 ОПК-7.1 ОПК-8.1 ОПК-2.2 ОПК-4.2 ОПК-7.2 ОПК-8.2 ОПК-2.3 ОПК-4.3 ОПК-7.3 ОПК-8.3 ПК-П2.1 ПК-П8.1 ПК-П2.2 ПК-П8.2 ПК-П2.3 ПК-П8.3

Вопросы/Задания:

1. Программная инженерия как дисциплина, комплексно охватывающая процесс разработки информационных систем

2. Структурное программирование.
3. Процедурно-ориентированное программирование.
4. Объектно-ориентированное программирование.
5. Модульное программирование.
6. Современные парадигмы программирования.
7. Унифицированный язык моделирования UML
8. Методология проектирования Microsoft.
9. Гибкие методологии разработки ПО: Scrum.
10. Гибкие методологии разработки ПО: Kanban.
11. Спецификация требований: состав и структура документа.
12. Техническое задание: назначение, основные разделы документа.
13. Технический проект: назначение, основные разделы документа.
14. Структура ТЭО.
15. Рабочий проект: назначение, основные разделы документа.
16. Программа тестовых испытаний.
17. Документирование требований – основы языка Gherkin.
18. Понятие и структура тестового скрипта.
19. Документация пользователя.
20. Принципы и инструменты документирования кода.
21. Тестирование как процесс обеспечения качества ПС: понятие, цели, виды тестирования.
22. Автоматизация тестирования.
23. Разработка через тестирование (TDD).
24. Понятие и структура тестового скрипта.
25. Тестовое покрытие.

26. Системы контроля версий. Git.
27. Опытнo-промышленная эксплуатация.
28. Стратегии развертывания системы.
29. Интеграционное тестирование системы.
30. Миграция данных.
31. Основные понятия ООП: инкапсуляция, наследование, полиморфизм.
32. Основы ООП: класс как абстрактный тип, конструкторы и деструкторы.
33. Основы ООП: объявление классов, объекты классов, указатель this.
34. Основы ООП: спецификаторы доступа класса.
35. Основы ООП: «друзья» классов (дружественные функции и классы).
36. Основы ООП: перегрузка и шаблоны функций.
37. Основы ООП: наследование классов, шаблоны классов.
38. Основы ООП: перегрузка стандартных операций.
39. Основы ООП: обработка исключительных ситуаций.
40. Унифицированный язык моделирования UML: цели создания и назначение.
41. Методология проектирования UML: диаграммы вариантов использования.
42. Методология проектирования UML: диаграммы состояний.
43. Методология проектирования UML: диаграммы деятельности.
44. Методология проектирования UML: диаграммы классов.
45. Методология проектирования UML: диаграммы последовательности.
46. Методология проектирования UML: диаграммы кооперации.
47. Методология проектирования UML: диаграммы компонентов.
48. Методология проектирования UML: диаграммы развертывания.
49. Методология проектирования Microsoft: схема слоев – назначение, пример.
50. Понятие и модели жизненного цикла программного средства.

51. Основные процессы жизненного цикла программного средства.
52. Гибкие методологии разработки ПО: сравнительный анализ, Scrumban, преимущества и недостатки.
53. Авторская и коллективная разработка.
54. Общинная модель разработки ПО. Основные отличия от коллективной разработки.
55. Коллективная разработка. Модель группы и иерархическая модель.
56. Коллективная разработка: роли и обязанности членов проектной группы. Деструктивное сочетание ролей.
57. Коллективная разработка: роли, деструктивное и созидательное сочетание ролей.
58. Управление требованиями: определение требований, управления требованиями, место и роль управления требованиями в процессе жизненного цикла ПО.
59. Управление требованиями: классификация требований.
60. Управление требованиями: методы сбора требований.
61. Проектирование и разработка интерфейса ПС: понятие, эргономика, типы диалогов.
62. Проектирование и разработка интерфейса ПС: раскадровка, макетирование.
63. Проектирование и разработка интерфейса ПС: принципы построения интерфейсов.
64. Практика работы с требованиями
описать не менее 5 историй пользователя при выявлении требований к программному продукту. Примеры программных продуктов: будильник, секундомер, блокнот, графический редактор, музыкальный проигрыватель, почтовый клиент, электронный учебник, банкомат, кинокасса, заказ такси, оплата картой, пульсометр
65. Практика проектирования UML
построить требуемую диаграмму UML заданной предметной области (согласно программному продукту). Варианты диаграмм UML: ВИ, последовательности, активности, состояний
66. Практика ООП
описать класс с заданными компонентными данными и функциями. Не менее 2х компонентных данных, минимум 1 конструктор, не менее 3 компонентных функций. Класс по программному продукту
67. Практика Agile
спланировать итерации проекта разработки программного продукта с учетом стандартных этапов жизненного цикла программного продукта. Длительность итерации принять равной 1 дню
68. Практика использования языка Gherkin
описать на языке Gherkin пользовательский сценарий авторизации аккаунта для использования программного продукта. Сценарий должен иметь 1 основной и 1 альтернативный поток

8. Материально-техническое и учебно-методическое обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Маран М. М. Программная инженерия / Маран М. М.. - 3-е изд., стер. - Санкт-Петербург: Лань, 2022. - 196 с. - 978-5-8114-9323-4. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/189470.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке
2. Турнецкая Е. Л. Программная инженерия. Интеграционный подход к разработке: учебник для вузов / Турнецкая Е. Л., Аграновский А. В.. - Санкт-Петербург: Лань, 2023. - 216 с. - 978-5-507-46898-0. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/352307.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке
3. ЕФАНОВА Н. В. Программная инженерия: учебник / ЕФАНОВА Н. В., Иванова Е. А.. - Краснодар: КубГАУ, 2022. - 174 с. - 978-5-907667-60-0. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=12688> (дата обращения: 21.06.2024). - Режим доступа: по подписке
4. Системная и программная инженерия / Миронов А. Н., Воронцов Ю. А., Михайлова Е. К., Трушин С. М.. - Москва: РТУ МИРЭА, 2022. - 129 с. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/310997.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке

Дополнительная литература

1. Леоненков, А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose: учебное пособие / А. В. Леоненков,. - Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. - 317 с. - 978-5-4497-0667-6. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/97554.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке
2. Достовалов Д. Н. Объектно-ориентированный анализ и проектирование. Задачи и примеры на C++: учебное пособие / Достовалов Д. Н., Лауферман О. В.. - Новосибирск: НГТУ, 2022. - 74 с. - 978-5-7782-4708-6. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/306206.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке
3. ЕФАНОВА Н. В. Программная инженерия: метод. рекомендации / ЕФАНОВА Н. В., Иванова Е. А.. - Краснодар: КубГАУ, 2020. - 35 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9081> (дата обращения: 02.05.2024). - Режим доступа: по подписке
4. Суханов, М. Б. Программная инженерия: учебное пособие / М. Б. Суханов,. - Программная инженерия - Санкт-Петербург: Санкт-Петербургский государственный университет промышленных технологий и дизайна, 2018. - 146 с. - 978-5-7937-1614-7. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/102465.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке
5. Липаев, В. В. Программная инженерия сложных заказных программных продуктов: учебное пособие / В. В. Липаев,. - Программная инженерия сложных заказных программных продуктов - Москва: МАКС Пресс, 2014. - 309 с. - 978-5-317-04750-4. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/27297.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке

8.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся

Профессиональные базы данных

Не используются.

Ресурсы «Интернет»

1. <http://edu.kubsau.ru/> - Образовательный портал КубГАУ
2. <http://www.iprbookshop.ru/> - IPRbook
3. <https://znanium.com/> - Znanium.com

8.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине позволяют:

- обеспечить взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействие посредством сети «Интернет»;
- фиксировать ход образовательного процесса, результатов промежуточной аттестации по дисциплине и результатов освоения образовательной программы;
- организовать процесс образования путем визуализации изучаемой информации посредством использования презентаций, учебных фильмов;
- контролировать результаты обучения на основе компьютерного тестирования.

Перечень лицензионного программного обеспечения:

- 1 Microsoft Windows - операционная система.
- 2 Microsoft Office (включает Word, Excel, Power Point) - пакет офисных приложений.

Перечень профессиональных баз данных и информационных справочных систем:

- 1 Гарант - правовая, <https://www.garant.ru/>
- 2 Консультант - правовая, <https://www.consultant.ru/>
- 3 Научная электронная библиотека eLibrary - универсальная, <https://elibrary.ru/>

Доступ к сети Интернет, доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения

(обновление производится по мере появления новых версий программы)

1. Dr.Web;
2. Консультант Плюс;
3. МойОфис;
4. ПО " 1С:Предприятие 8.3 ПРОФ. 1С:Предприятие. Облачная подсистема Фреш ";
5. Гарант;
6. Система тестирования INDIGO;
7. Microsoft Windows Professional 10 (посредством апгрейда лицензии Microsoft Windows Professional 8.1 ;
8. Microsoft Windows 7 Professional 64 bit;
9. Microsoft Windows 7 Professional 64 bit;
10. 1С:Предпр.8.Комплект для обучения в высших и средних учебных заведениях;

Перечень информационно-справочных систем

(обновление выполняется еженедельно)

Не используется.

8.4. Специальные помещения, лаборатории и лабораторное оборудование

Университет располагает на праве собственности или ином законном основании материально-техническим обеспечением образовательной деятельности (помещениями и оборудованием) для реализации программы бакалавриата, специалитета, магистратуры по Блоку 1 "Дисциплины (модули)" и Блоку 3 "Государственная итоговая аттестация" в соответствии с учебным планом.

Каждый обучающийся в течение всего периода обучения обеспечен индивидуальным неограниченным доступом к электронной информационно-образовательной среде университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории университета, так и вне его. Условия для функционирования электронной информационно-образовательной среды могут быть созданы с использованием ресурсов иных организаций.

Компьютерный класс

418эл

Доска классная дк 12Э2410 - 1 шт.

Компьютер персональный iRU/8Гб/512Гб - 1 шт.

стол аудиторный - 0 шт.

Стул жесткий - 26 шт.

420эл

доска ДК11Э3010(мел) - 1 шт.

Компьютер персональный iRU/8Гб/512Гб - 1 шт.

стол аудиторный - 0 шт.

стол компьютерный - 1 шт.

Стул жесткий - 26 шт.

9. Методические указания по освоению дисциплины (модуля)

Учебная работа по направлению подготовки осуществляется в форме контактной работы с преподавателем, самостоятельной работы обучающегося, текущей и промежуточной аттестаций, иных формах, предлагаемых университетом. Учебный материал дисциплины структурирован и его изучение производится в тематической последовательности. Содержание методических указаний должно соответствовать требованиям Федерального государственного образовательного стандарта и учебных программ по дисциплине. Самостоятельная работа студентов может быть выполнена с помощью материалов, размещенных на портале поддержки Moodle.

Методические указания по формам работы

Лекционные занятия

Передача значительного объема систематизированной информации в устной форме достаточно большой аудитории. Дает возможность экономно и систематично излагать учебный материал. Обучающиеся изучают лекционный материал, размещенный на портале поддержки обучения Moodle.

Лабораторные занятия

Практическое освоение студентами научно-теоретических положений изучаемого предмета, овладение ими техникой экспериментирования в соответствующей отрасли науки. Лабораторные занятия проводятся с использованием методических указаний, размещенных на образовательном портале университета.

Описание возможностей изучения дисциплины лицами с ОВЗ и инвалидами

Для инвалидов и лиц с ОВЗ может изменяться объём дисциплины (модуля) в часах, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося (при этом не увеличивается количество зачётных единиц, выделенных на освоение дисциплины).

Фонды оценочных средств адаптируются к ограничениям здоровья и восприятия информации обучающимися.

Основные формы представления оценочных средств – в печатной форме или в форме электронного документа.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением зрения:

- устная проверка: дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.;
- с использованием компьютера и специального ПО: работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, дистанционные формы, если позволяет острота зрения - графические работы и др.;
- при возможности письменная проверка с использованием рельефно-точечной системы Брайля, увеличенного шрифта, использование специальных технических средств (тифлотехнических средств): контрольные, графические работы, тестирование, домашние задания, эссе, отчеты и др.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением слуха:

- письменная проверка: контрольные, графические работы, тестирование, домашние задания, эссе, письменные коллоквиумы, отчеты и др.;
- с использованием компьютера: работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, графические работы, дистанционные формы и др.;
- при возможности устная проверка с использованием специальных технических средств (аудиосредств, средств коммуникации, звукоусиливающей аппаратуры и др.): дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением опорно-двигательного аппарата:

- письменная проверка с использованием специальных технических средств (альтернативных средств ввода, управления компьютером и др.): контрольные, графические работы, тестирование, домашние задания, эссе, письменные коллоквиумы, отчеты и др.;
- устная проверка, с использованием специальных технических средств (средств коммуникаций): дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.;
- с использованием компьютера и специального ПО (альтернативных средств ввода и управления компьютером и др.): работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, графические работы, дистанционные формы предпочтительнее обучающимся, ограниченными в передвижении и др.

Адаптация процедуры проведения промежуточной аттестации для инвалидов и лиц с ОВЗ.

В ходе проведения промежуточной аттестации предусмотрено:

- предъявление обучающимся печатных и (или) электронных материалов в формах, адаптированных к ограничениям их здоровья;
- возможность пользоваться индивидуальными устройствами и средствами, позволяющими адаптировать материалы, осуществлять приём и передачу информации с учетом их индивидуальных особенностей;
- увеличение продолжительности проведения аттестации;
- возможность присутствия ассистента и оказания им необходимой помощи (занять рабочее место, передвигаться, прочитать и оформить задание, общаться с преподавателем).

Формы промежуточной аттестации для инвалидов и лиц с ОВЗ должны учитывать индивидуальные и психофизические особенности обучающегося/обучающихся по АООП ВО (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с

нарушениями зрения:

- предоставление образовательного контента в текстовом электронном формате, позволяющем переводить плоскочечатную информацию в аудиальную или тактильную форму;
- возможность использовать индивидуальные устройства и средства, позволяющие адаптировать материалы, осуществлять приём и передачу информации с учетом индивидуальных особенностей и состояния здоровья студента;
- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
- использование чёткого и увеличенного по размеру шрифта и графических объектов в мультимедийных презентациях;
- использование инструментов «лупа», «прожектор» при работе с интерактивной доской;
- озвучивание визуальной информации, представленной обучающимся в ходе занятий;
- обеспечение раздаточным материалом, дублирующим информацию, выводимую на экран;
- наличие подписей и описания у всех используемых в процессе обучения рисунков и иных графических объектов, что даёт возможность перевести письменный текст в аудиальный;
- обеспечение особого речевого режима преподавания: лекции читаются громко, разборчиво, отчётливо, с паузами между смысловыми блоками информации, обеспечивается интонирование, повторение, акцентирование, профилактика рассеивания внимания;
- минимизация внешнего шума и обеспечение спокойной аудиальной обстановки;
- возможность вести запись учебной информации студентами в удобной для них форме (аудиально, аудиовизуально, на ноутбуке, в виде пометок в заранее подготовленном тексте);
- увеличение доли методов социальной стимуляции (обращение внимания, апелляция к ограничениям по времени, контактные виды работ, групповые задания и др.) на практических и лабораторных занятиях;
- минимизирование заданий, требующих активного использования зрительной памяти и зрительного внимания;
- применение поэтапной системы контроля, более частый контроль выполнения заданий для самостоятельной работы.

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями опорно-двигательного аппарата (маломобильные студенты, студенты, имеющие трудности передвижения и патологию верхних конечностей):

- возможность использовать специальное программное обеспечение и специальное оборудование и позволяющее компенсировать двигательное нарушение (коляски, ходунки, трости и др.);
- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
- применение дополнительных средств активизации процессов запоминания и повторения;
- опора на определенные и точные понятия;
- использование для иллюстрации конкретных примеров;
- применение вопросов для мониторинга понимания;
- разделение изучаемого материала на небольшие логические блоки;
- увеличение доли конкретного материала и соблюдение принципа от простого к сложному при объяснении материала;
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- увеличение доли методов социальной стимуляции (обращение внимания, апелляция к ограничениям по времени, контактные виды работ, групповые задания др.);
- обеспечение беспрепятственного доступа в помещения, а также пребывания них;
- наличие возможности использовать индивидуальные устройства и средства, позволяющие обеспечить реализацию эргономических принципов и комфортное пребывание на месте в течение всего периода учёбы (подставки, специальные подушки и др.).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями слуха (глухие, слабослышащие, позднооглохшие):

- предоставление образовательного контента в текстовом электронном формате, позволяющем переводить аудиальную форму лекции в плоскостную информацию;
 - наличие возможности использовать индивидуальные звукоусиливающие устройства и сурдотехнические средства, позволяющие осуществлять приём и передачу информации; осуществлять взаимообратный перевод текстовых и аудиофайлов (блокнот для речевого ввода), а также запись и воспроизведение зрительной информации;
 - наличие системы заданий, обеспечивающих систематизацию вербального материала, его схематизацию, перевод в таблицы, схемы, опорные тексты, глоссарий;
 - наличие наглядного сопровождения изучаемого материала (структурно-логические схемы, таблицы, графики, концентрирующие и обобщающие информацию, опорные конспекты, раздаточный материал);
 - наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
 - обеспечение практики опережающего чтения, когда студенты заранее знакомятся с материалом и выделяют незнакомые и непонятные слова и фрагменты;
 - особый речевой режим работы (отказ от длинных фраз и сложных предложений, хорошая артикуляция; четкость изложения, отсутствие лишних слов; повторение фраз без изменения слов и порядка их следования; обеспечение зрительного контакта во время говорения и чуть более медленного темпа речи, использование естественных жестов и мимики);
 - чёткое соблюдение алгоритма занятия и заданий для самостоятельной работы (называние темы, постановка цели, сообщение и запись плана, выделение основных понятий и методов их изучения, указание видов деятельности студентов и способов проверки усвоения материала, словарная работа);
 - соблюдение требований к предъявляемым учебным текстам (разбивка текста на части; выделение опорных смысловых пунктов; использование наглядных средств);
 - минимизация внешних шумов;
 - предоставление возможности соотносить вербальный и графический материал; комплексное использование письменных и устных средств коммуникации при работе в группе;
 - сочетание на занятиях всех видов речевой деятельности (говорения, слушания, чтения, письма, зрительного восприятия с лица говорящего).
- Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с прочими видами нарушений (ДЦП с нарушениями речи, заболевания эндокринной, центральной нервной и сердечно-сосудистой систем, онкологические заболевания):
- наличие возможности использовать индивидуальные устройства и средства, позволяющие осуществлять приём и передачу информации;
 - наличие системы заданий, обеспечивающих систематизацию вербального материала, его схематизацию, перевод в таблицы, схемы, опорные тексты, глоссарий;
 - наличие наглядного сопровождения изучаемого материала;
 - наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
 - обеспечение практики опережающего чтения, когда студенты заранее знакомятся с материалом и выделяют незнакомые и непонятные слова и фрагменты;
 - предоставление возможности соотносить вербальный и графический материал; комплексное использование письменных и устных средств коммуникации при работе в группе;
 - сочетание на занятиях всех видов речевой деятельности (говорения, слушания, чтения, письма, зрительного восприятия с лица говорящего);
 - предоставление образовательного контента в текстовом электронном формате;
 - предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
 - возможность вести запись учебной информации студентами в удобной для них форме (аудиально, аудиовизуально, в виде пометок в заранее подготовленном тексте);
 - применение поэтапной системы контроля, более частый контроль выполнения заданий для самостоятельной работы;
 - стимулирование выработки у студентов навыков самоорганизации и самоконтроля;

– наличие пауз для отдыха и смены видов деятельности по ходу занятия.

10. Методические рекомендации по освоению дисциплины (модуля)

Дисциплина "Программная инженерия" ведётся в соответствии с календарным учебным планом и расписанием занятий по неделям. Темы проведения занятий определяются тематическим планом рабочей программы дисциплины.