

Государственное образовательное учреждение
**«ПРИДНЕСТРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. Т.Г. Шевченко»**
Рыбницкий филиал

Кафедра «Информатика и программная инженерия»

УТВЕРЖДАЮ:

Заведующий кафедрой

Тигульская Л.А., доцент
«25» сентября 2021 г.

Фонд оценочных средств

«Проектирование программного обеспечения»

Направление подготовки:

09.03.04 «Программная инженерия»

Профиль подготовки:

«Разработка программно-информационных систем»

Квалификация (степень) выпускника

Бакалавр

Форма обучения:

очная

Год набора

2018

Разработчик: ст. преподаватель

Сташкова О.В. «24» сентября 2021 г.

Рыбница, 2021

ПАСПОРТ

фонда оценочных средств по учебной дисциплине

1. В результате изучения дисциплины «Проектирование программного обеспечения» у обучающихся должны быть сформированы следующие компетенции:

Знать:

- жизненный цикл программ, оценку качества программных продуктов, технологии разработки программных комплексов, CASE-средства;
- методы и алгоритмы объектно-ориентированного программирования;
- планирование и управление проектами;
- тестирование и обеспечение качества;
- работу в коллективе, организацию команды разработчиков;
- документацию;
- сопровождение;
- стандарты качества программного обеспечения: ISO 9000, CASE-средства разработки;
- структурное проектирование;
- восходящий и нисходящий способы проектирования и реализации программного обеспечения;
- встроенные системы синтаксического анализа.

Уметь:

- использовать типовые программные продукты, ориентированные на решение научных, проектных и технологических задач;
- самостоятельно приобретать необходимые знания из предметной области;
- проводить анализ предметной области;
- составлять техническое задание, спецификации требований;
- выбирать необходимые математические модели и способы их алгоритмической реализации;
- осуществлять выбор программных и инструментальных средств для разработки, создания и отладки программного обеспечения;
- организовывать верификацию, тестирование и проверку стабильности программного обеспечения;
- осуществлять разработку пользовательского интерфейса и интеграцию проекта.

Владеть:

- технологией разработки программного обеспечения по программе подготовки.

2. Программа оценивания контролируемой компетенции:

Текущая аттестация	Контролируемые модули, разделы (темы) дисциплины и их наименование	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	Проблемы разработки сложных программ.	ОПК-3, ОПК-4	Вопросы для устного опроса
2	Унифицированный язык моделирования UML.	ПК-12, ПК-15, ПК-20, ПК-21	Тестовые задания для промежуточного контроля
Промежуточная аттестация		Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1		ОПК-3, ОПК-4, ПК-12, ПК-15, ПК-20, ПК-21	Итоговой тест Комплект КИМ

«УТВЕРЖДАЮ»

зав. кафедрой ИиПИ,

доцент _____ Л. А. Тягульская

« ____ » _____ 2021 г.

Вопросов для устного опроса по дисциплине
«Проектирование программного обеспечения»
для студентов IV курса (з/о)
направления «Программная инженерия» профиля подготовки
«Разработка программно-информационных систем»

1. Цели и задачи технологий разработки ПО. Особенности современных крупных проектов.
2. Основные определения. Программные средства. Программное обеспечение (ПО). Программный продукт. Проектирование ПО. Программирование. Классификация типов программного обеспечения.
3. Составные части технологии программирования. Проект, продукт, процесс и персонал.
4. Основные понятия технологии программирования. Процессы и модели. Фазы и витки.
5. Выявление и анализ требований. Требования к программному обеспечению. Схема разработки требований. Управление требованиями.
6. Свойства требований (способы устранения неоднозначности; ошибки, нарушающие полноту; ситуации, нарушающие целостность; способы улучшения целостности; упорядоченность; проверяемость; изменяемость; прослеживаемость).
7. Способы выражения (записи) требований в ТЗ (варианты использования; диаграмма потоков данных; диаграмма перехода состояний).
8. Общий шаблон ТЗ в стандарте IEEE 830-1993 (требования заказчика; детальные требования; принципы составления и способы их организации).
9. Архитектурное и детальное проектирование.
10. Реализация и кодирование.
11. Тестирование и верификация. Процесс контроля качества. Методы контроля качества.
12. Цели тестирования. Верификация, валидация и системное тестирование.
13. Характеристики качества и критерии качества ПО, (надежность; эффективность; практичность; универсальность; сопровождаемость; корректность; обеспечение завершенности ПС).
14. Жизненный цикл программы. Циклический характер разработки.
15. Процессы и модели. Какой международный стандарт определяет перечень и содержание процессов ЖЦ ПО?
16. Основные группы процессов жизненного цикла и процессы каждой из групп.
17. Стадии жизненного цикла ПС, особенности разработки ПС, основные международные стандарты, пять подходов к разработке.
18. Модели процесса разработки. Водопадный подход (каскадная модель), выводы о применимости классической каскадной модели.
19. Модели процесса разработки. Итерационный подход (спиральная и инкрементальная модели). Гибкие модели процесса разработки.
20. Международные стандарты проектирования, разработки, оформления документации, пользовательского интерфейса ПИ.
21. Измерения, меры и метрики. Размерно-ориентированные метрики. Функционально-ориентированные метрики.
22. Выполнение оценки проекта на основе LOC- и FP-метрик.
23. Коллективный характер разработки. Состав и структура коллектива разработчиков, их функции. Конструирование модели команды.
24. Конструирование модели процесса. Выявление требований к процессу (спецификация требований). Техническое задание. Подходы к разработке технического задания.

25. Планирование проекта. Уточнение содержания и состава работ. Планирование управления содержанием.

26. Планирование организационной структуры и планирование управления конфигурациями.

27. Планирование управления качеством. Базовое расписание проекта.

28. Парадигмы программирования.

29. Структурное программирование.

30. Логическое программирование.

31. Объектно-ориентированное программирование

32. Программная архитектура. Событийное управление.

33. Понятие декомпозиции, классификация базовых архитектур (архитектуры потоков данных; архитектуры независимых компонентов; архитектуры виртуальных машин; уровневые архитектуры.

34. Архитектура клиент/сервер. Службы.

35. Трехслойная архитектура.

36. Проектирование программ. Концептуальное проектирование.

37. Логическое проектирование. Детальное проектирование.

38. Кодирование. Программирование по образцу. Образцы проектирования.

39. Доказательное программирование. Программирование вширь. Форматирование кода

40. Тестирование и отладка. Критерии приемлемости. Виды тестирования. Методы отладки.

41. Инструментальные средства проектирования. Системы автоматизации разработки программных систем.

42. Сертификация фирм разработчиков по модели качества СММ.

43. Документация, создаваемая в процессе разработки программных средств. Документы управления разработкой ПС. Документы, входящие в состав ПС.

44. Пользовательская документация.

45. Документация по сопровождению программных средств.

46. Человеческий фактор в управлении проектами. Задача n-личностей. Закон Брукса. Подходы к управлению группами и руководству ими.

Ст. преподаватель,  О.В. Сташкова

«УТВЕРЖДАЮ»

зав. кафедрой ИиПИ,

доцент _____ Л. А. Тягульская

« ____ » _____ 2021 г.

Тестовые задания для промежуточного контроля знаний
по дисциплине «Проектирование программного обеспечения»
для студентов IV курса (з/о)
направления «Программная инженерия» профиля подготовки
«Разработка программно-информационных систем»

Лабораторная работа №1

Тема: Унифицированный язык моделирования (UML). Среда Rational Rose.

Вопрос №1

Какие из нижеперечисленных элементов языка UML относятся к основополагающим?

- a) сущность
- b) актер
- c) отношение
- d) диаграмма

Вопрос №2

Назовите основное назначение языка моделирования UML.

- a) отображение архитектуры разрабатываемых программных средств
- b) обеспечение взаимопонимания между заказчиками и разработчиками
- c) создание моделей информационных систем и программного обеспечения
- d) создание программного обеспечения

Вопрос №3

Дайте точное определение понятию *сущность* - ...

- a) связанный граф в информационной системе
- b) основной способ связи между абстракциями
- c) модель программного обеспечения
- d) концептуальный или физический элемент системы

Вопрос №4

Какие виды сущностей Вам известны?

- a) структурные
- b) поведенческие
- c) группирующие
- d) аннотационные

Вопрос №5

Что собой представляет диаграмма компонентов?

- a) связанный граф в информационной системе
- b) основной способ связи между абстракциями
- c) модель программного обеспечения
- d) концептуальный или физический элемент системы

Вопрос №6

Какие из нижеперечисленных диаграмм существуют?

- a) диаграмма зависимостей
- b) диаграмма состояний
- c) диаграмма классов
- d) диаграмма поведения

Вопрос №7

Дайте точное определение понятию *отношение* - ...

- a) связанный граф в информационной системе
- b) основной способ связи между абстракциями
- c) модель программного обеспечения
- d) концептуальный или физический элемент системы

Вопрос №8

Приведите примеры отношений, которые Вам известны.

- a) зависимость
- b) обобщение
- c) ассоциация
- d) реализация
- e) ни один из вариантов не является правильным

Вопрос №9

Для чего служит браузер среды Rational Rose?

- a) для навигации по разрабатываемой модели
- b) для визуализации различных представлений модели проекта
- c) для документирования элементов модели Rose
- d) для автоматической записи различной служебной информации

Вопрос №10

Перечислите основные элементы главного окна программы Rational Rose.

- a) актеры, ребра, графы
- b) сущности, отношения, диаграммы
- c) браузер, окно документации, окно диаграмм, журнал
- d) зависимость, ассоциация, состояние

Лабораторная работа №2

Тема: Диаграммы Вариантов Использования (Use Case).

Вопрос №1

Для чего предназначены диаграммы вариантов использования (Use Case)?

- a) для отображения взаимодействия между некоторыми объектами
- b) для обозначения множества объектов, которые обладают одинаковой структурой
- c) для создания списков операций, которые может выполнять система
- d) для указания зависимостей между сущностями

Вопрос №2

Кто или что из нижеперечисленного является действующими лицами в диаграмме вариантов использования?

- a) прецеденты
- b) актеры
- c) отношения
- d) диаграммы

Вопрос №3

Что необходимо учитывать при разработке диаграмм вариантов использования?

- a) границы предметной области
- b) способы взаимодействия между основными элементами системы
- c) требования к функциональному поведению
- d) документация

Вопрос №4

Выделите основные элементы диаграмм вариантов использования.

- a) актер
- b) спецификация
- c) документация
- d) связь

Вопрос №5

Укажите правильное назначение понятия связь - ...

- a) указание взаимодействия между актерами и вариантами использования
- b) выделение основных функций системы
- c) определение зависимостей между вариантами использования
- d) дополнение к документации

Вопрос №6

- Для чего в диаграмме вариантов использования применяется элемент «актер(действующее лицо)»?
- а) для моделирования внешних по отношению к системе сущностей
 - б) для определения главных и второстепенных связей
 - в) для описания основных задач системы
 - г) элемент «Актор» не применяется в диаграмме вариантов использования

Вопрос №7

Какие актеры называются *абстрактными*?

- а) актеры, взаимодействующие с системой и использующие ее в качестве отдельных пользователей
- б) актеры, использующиеся для приема/передачи сообщений
- в) актеры, не имеющие дополнительных экземпляров
- г) актеры, использующиеся для описания основных задач системы

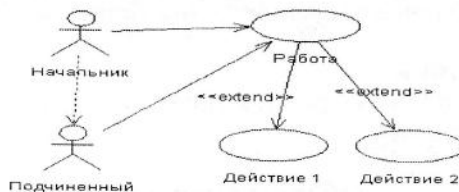
Вопрос №8

Назовите основное назначение окна «Спецификация»?

- а) документирование элемента
- б) задание основных свойств и стереотипа
- в) указание взаимосвязи между экземплярами вариантов использования
- г) для обозначения множества объектов, которые обладают одинаковой структурой

Вопрос №9

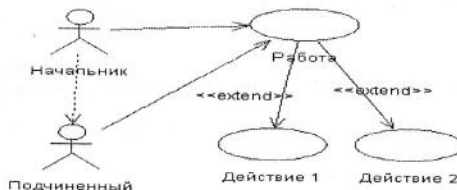
На рисунке представлен один из вариантов диаграмм вариантов использования. Какая связь имеет место между объектами «начальник» и «подчиненный»?



- а) коммуникация
- б) обобщение
- в) зависимость
- г) реализация

Вопрос №10

Что обозначает надпись «*extend*» на примере связей между объектами «работа» и «действие1»?



- а) расширение
- б) использование
- в) наследование
- г) реализация

Лабораторная работа №3

Тема: Диаграммы взаимодействий. Диаграммы последовательности и сотрудничества.

Вопрос №1

Выберите из предложенных вариантов, для чего используются диаграммы взаимодействий?

- а) для моделирования взаимодействия объектов
- б) для указания зависимостей между сущностями
- в) для создания списков операций, которые может выполнять система
- г) для обозначения множества объектов, которые обладают одинаковой структурой

Вопрос №2

Сколько измерений характерно для диаграммы последовательности?

- а) одно
- б) два
- в) три
- г) четыре

Вопрос №3

Какое предназначение имеет термин «*линия жизни*»?

- а) установка связей между объектами информационной системы

- b) указание зависимостей между сущностями
- c) определение последовательности между функциями системы
- d) обозначение времени существования объекта в системе

Вопрос №4

Укажите основное назначение термина «фокус управления»?

- a) указание зависимостей между сущностями
- b) создание списков операций, которые может выполнять система
- c) обозначение периода начала и конца активности объекта в системе
- d) обозначение времени существования объекта в системе

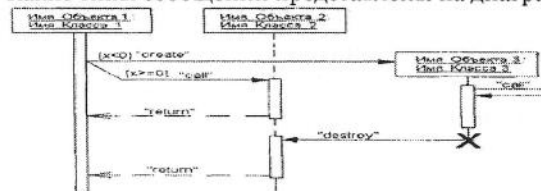
Вопрос №5

Дайте определение понятию «сообщение»?

- a) сигнал начала периода активности объекта
- b) сигнал окончания периода активности объекта
- c) данные о существовании объекта в системе
- d) фрагмент информации между объектами

Вопрос №6

Какие типы сообщений представлены на диаграмме?



- a) вызов
- b) возврат
- c) уничтожение
- d) создание

Вопрос №7

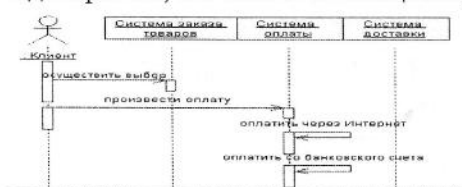
Определите на представленной диаграмме, какое действие будет происходить третьим?



- a) осуществить выбор
- b) произвести оплату
- c) оплатить через Интернет
- d) оплатить через банк

Вопрос №8

Определите на представленной диаграмме, кто является инициатором работы системы?



- a) система заказа товаров
- b) клиент
- c) система оплаты
- d) система доставки

Вопрос №9

Укажите основное назначение диаграмм кооперации.

- a) раскрытие информации о работе системы
- b) раскрытие информации о внутреннем устройстве системы
- c) описание динамического аспекта взаимодействия объектов
- d) описание статического аспекта взаимосвязи объектов

Вопрос №10

Какие стереотипы связей характерны для диаграммы кооперации.

- a) ассоциация
- b) параметр метода
- c) локальная переменная метода
- d) глобальная переменная
- e) рефлексивная связь
- f) все варианты

Лабораторная работа № 4.

Тема: Диаграммы классов. Шаблоны. Работа с пакетами.

Вопрос №1

Какое из определений понятия *класс* является правильным?

- a) сущность, описывающая множество объектов со сходной структурой, поведением и связями
- b) группа объектов в рамках одного варианта использования
- c) граф, состоящий из объектов и соединяющих дуг с обозначенными на них событиями
- d) все варианты

Вопрос №2

С какой целью в языке UML используется понятие *диаграмма классов*?

- a) выделение сущностей в качестве отдельных классов
- b) разбиение системы на отдельные классы для упрощения восприятия
- c) обозначение классов со схожей структурой, поведением и отношениями
- d) разбиение на отдельные модули для дальнейшей реализации

Вопрос №3

Какие элементы классов системы являются обязательными?

- a) имя
- b) атрибуты
- c) функции
- d) все варианты

Вопрос №4

Приведите примеры известных Вам стереотипов классов.

- a) граница
- b) сущность
- c) пакет
- d) управление

Вопрос №5

Какие примеры признаков видимости элементов класса Вам известны?

- a) пакетный
- b) общий
- c) защищенный
- d) закрытый

Вопрос №6

Что представляет собой ассоциация класса?

- a) связь между экземплярами
- b) область видимости экземпляров
- c) указание зависимости потомка и родителя
- d) абстрактный тип класса

Вопрос №7

Что собой *представляет* атрибут класса?

- a) элемент данных, содержащийся в объекте, принадлежащем описываемому классу
- b) сущность, определяющая некое действие, которое может быть выполнено представителем класса
- c) возможное количество объектов, которые могут быть связаны с ассоциацией
- d) все варианты

Вопрос №8

В чем отличие атрибутов от ассоциаций?

- a) данные элементы эквивалентны
- b) атрибут предполагает навигацию 1...*
- c) атрибут предполагает навигацию 1...1

d) ни один из предложенных вариантов не является правильным

Вопрос №9

Что представляет собой операция класса?

- a) процесс, реализуемый классом
- b) действие системы в определенный промежуток времени
- c) указание признака видимости
- d) функция информационного объекта

Вопрос №10

Дайте определение понятию *пакет*?

- a) общие атрибуты/свойства какого-либо класса
- b) хранилище набора классов или других пакетов
- c) признак видимости класса
- d) информационный объект системы

Ответы

1 лабораторная		2 лабораторная		3 лабораторная		4 лабораторная	
№ вопроса	Ответ	№ вопроса	Ответ	№ вопроса	Ответ	№ вопроса	Ответ
1	a,c,d	1	c	1	a	1	a
2	c	2	b	2	b	2	c
3	d	3	a,c,d	3	d	3	a
4	a,b,c,d	4	a,b,d	4	c	4	a,b
5	a	5	a	5	d	5	a,b,c,d
6	b,c	6	a	6	a,b,c, d	6	a
7	b	7	c	7	c	7	a
8	a,b,c,d	8	a,b	8	b	8	c
9	a	9	c	9	c	9	a
10	c	10	a	10	f	10	b

Ст. преподаватель, _____ О.В. Сташкова

«УТВЕРЖДАЮ»

зав. кафедрой ИиПИ,

доцент _____ Л. А. Тягульская

«___» _____ 2021 г.

Итоговый тест по дисциплине
«Проектирование программного обеспечения»
для студентов IV курса (з/о)
направления «Программная инженерия» профиля подготовки
«Разработка программно-информационных систем»

1. Программа или логически связанная совокупность программ обработки данных на носителях данных, снабжённая необходимой программной документацией, называется ...
 - a) CASE- технологии.
 - b) Программным средством.**
 - c) Парадигма программирования.
 - d) Утилита.
2. ... технологии разработки программного обеспечения является программа, эффективно и надёжно выполняющая требуемые функции на реальных компьютерах.
 - a) Предметом
 - b) Продуктом.**
 - c) Объектом.
 - d) Субъектом.
3. ... - это совокупность процессов, которая отражает его различные состояния, начиная с момента принятия решения о необходимости создания программного средства и заканчивая его полным изъятием из эксплуатации.
 - a) Технология разработки ПС.
 - b) Жизненный цикл.**
 - c) CASE- технологии.
 - d) Аттестация ПС.
4. Укажите, на какой группе процессов не базируется структура жизненного цикла (согласно стандартом ISO/IEC 12207):
 - a) Основные процессы.
 - b) Эргономические процессы.**
 - c) Вспомогательные процессы.
 - d) Организационные процессы.
5. Какие процессы жизненного цикла относятся к вспомогательным:
 - a) Приобретение, поставка, разработка, эксплуатация, сопровождение.
 - b) Управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого жизненного цикла, обучение.
 - c) Документирование, обеспечение качества, управление конфигурацией, верификация, аттестация.**
6. Какие процессы жизненного цикла относятся к организационным:
 - a) Приобретение, поставка, разработка, эксплуатация, сопровождение.

- b) **Управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого жизненного цикла, обучение.**
 - c) Документирование, обеспечение качества, управление конфигурацией, верификация, аттестация.
7. Какие процессы жизненного цикла относятся к основным:
- a) **Приобретение, поставка, разработка, эксплуатация, сопровождение.**
 - b) Управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого жизненного цикла, обучение.
 - c) Документирование, обеспечение качества, управление конфигурацией, верификация, аттестация.
8. Что не является стадией ЖЦ:
- a) Стадия разработки.
 - b) Стадия производство программного изделия.
 - c) **Стадия аттестация.**
 - d) Стадия эксплуатации.
9. Укажите этапы стадии разработки ЖЦ:
- a) **Этап внешнего описания.**
 - b) Этап документирования.
 - c) **Этап конструирования.**
 - d) **Этап кодирования.**
 - e) **Этап аттестации.**
 - f) Этап производства.
 - g) Этап эксплуатации.
10. ... ПС охватывает процессы: разработку архитектуры, разработку структур программ и данных, а также их детальную спецификацию.
- a) Кодирование.
 - b) Верификация.
 - c) **Конструирование.**
11. Экземпляр или копия, снятая с разработанного ПС называется
- a) Программной версией.
 - b) Программным средством.
 - c) **Программным изделием.**
12. Стадия эксплуатации представлена совокупностью фаз. Укажите те фазы, которые не относятся к данной стадии.
- a) **Фаза аттестации.**
 - b) Фаза эксплуатации.
 - c) Фаза сопровождения.
 - d) **Фаза сертификации.**
13. Под моделью жизненного цикла программного средства понимается ...
- a) **структура, определяющая последовательность выполнения стадий, и взаимосвязи стадий на протяжении жизненного цикла;**
 - b) логически завершённые части жизненного цикла;
 - c) совокупность процессов характеризующаяся определёнными задачами и методами их решения, исходными данными и результатами.
14. Укажите модели жизненного цикла:
- a) **Каскадная модель.**

- b) Древовидная модель.
 - c) **Спиралевидная модель.**
 - d) V-образная модель.
 - e) Циклическая модель.
15. Rational Unified Process (RUP) - это ...
- a) **Методология создания программных средств.**
 - b) Технология проектирования пользовательского интерфейса.
 - c) Парадигма программирования.
16. Какое понятие не включает предмет Rational Unified Process (RUP) :
- a) Персонал.
 - b) Проект.
 - c) Продукт.
 - d) Утилиты.
 - e) **Стандарты.**
 - f) Процесс.
17. К отличительным чертам унифицированного процесса разработки программного обеспечения относятся:
- a) **Итеративный подход к разработке.**
 - b) **Функции как основа проекта.**
 - c) **Управление требованиями и изменениями требований.**
 - d) **Приоритет архитектуры.**
 - e) **Способность к изменению конфигурации.**
 - f) **Визуальное моделирование.**
 - g) **Объектно- и сервисно-ориентированное проектирование.**
18. ... - это совокупность его черт и характеристик, которые влияют на его способность удовлетворять заданным потребностям пользователей.
- a) Сертифицированность ПС.
 - b) **Качество ПС.**
 - c) Функциональность ПС.
19. ... - способность ПС выполнять набор функций, удовлетворяющих заданным потребностям пользователей.
- d) Надёжность.
 - e) Завершённость.
 - f) Мобильность.
 - g) **Функциональность.**
20. ... - это способность ПС с достаточно большой вероятностью безотказно выполнять определённые функции при заданных условиях в течение заданного периода времени.
- a) **Надёжность.**
 - b) Завершённость.
 - c) Мобильность.
 - d) Функциональность.
21. это способность ПС быть перенесённым из одной среды (окружения) в другую, в частности, с одного компьютера на другой.
- a) Надёжность.
 - b) Завершённость.
 - c) **Мобильность.**

- d) **Функциональность.**
22. В каком документе фиксируются требования к качеству программного средства:
- a) Функциональная спецификация.
 - b) **Нефункциональная спецификация.**
 - c) Сертификат качества.
23. ... описывает поведение программного средства и определяет функции, которые должно выполнять ПС
- a) **Функциональная спецификация.**
 - b) Нefункциональная спецификация.
 - c) Сертификат качества.
 - d) Требования к процессу разработки.
24. ... - мера, характеризующая приемлемость величины погрешности в выдаваемых программами ПС результатах с точки зрения предполагаемого их использования.
- e) Автономность.
 - f) Устойчивость.
 - g) **Точность.**
 - h) Защищённость.
 - i) Документированность по применению.
25. ... - свойство, характеризующее способность ПС выполнять предписанные функции без помощи других компонент программного обеспечения.
- a) **Автономность.**
 - b) Устойчивость.
 - c) Точность.
 - d) Защищённость.
 - e) Документированность по применению.
26. ... - свойство, характеризующее способность ПС противостоять преднамеренным или нечаянным деструктивным (разрушающим) действиям пользователя.
- a) Автономность.
 - b) Устойчивость.
 - c) Точность.
 - d) **Защищённость.**
 - e) Документированность по применению.
27. ... - свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на задание неправильных (ошибочных) входных данных.
- a) Автономность.
 - b) **Устойчивость.**
 - c) Точность.
 - d) Защищённость.
 - e) Документированность по применению.
28. ... - свойство, характеризующее степень, в которой ПС облегчает задание или описание входных данных, а также обеспечивает выдачу полезных сведений в форме и с содержанием, простыми для понимания.
- a) Автономность.
 - b) Коммуникабельность.
 - c) Точность.
 - d) Защищённость.

- е) Документированность по применению.
29. Функциональная спецификация состоит из трех частей. Укажите лишнюю.
- а) Описание внешней информационной среды, к которой должны применяться программы разрабатываемой ПС.
 - б) Определение функций ПС, определённых на множестве состояний этой информационной среды (такие функции называют внешними функциями ПС).
 - в) **Описание тех достаточно элементарных свойств, которые требуется обеспечить в разрабатываемом ПС, и которые в совокупности образуют приемлемое для пользователя качество ПС.**
 - г) Описание нежелательных (исключительных) ситуаций, которые могут возникнуть при выполнении программ ПС, и реакций на эти ситуации, которые должны обеспечить соответствующие программы.
30. Укажите метод, который не относится к методам контроля спецификаций требований.
- а) Статический просмотр.
 - б) Смежный контроль.
 - в) Пользовательский контроль.
 - г) **Контроль CASE-средствами.**
 - е) Ручная имитация.

Ст. преподаватель,  О.В. Сташкова

«УТВЕРЖДАЮ»

зав. кафедрой ИиПИ,

доцент _____ Л. А. Тягульская

« ____ » _____ 2021 г.

Вопросы к экзамену по дисциплине
«Проектирование программного обеспечения»
для студентов IV курса (з/о)
направления «Программная инженерия» профиля подготовки
«Разработка программно-информационных систем»

1. Цели и задачи технологий разработки ПО. Основные определения. Программные средства.
2. Внешнее описание ПС: назначение и структура.
3. Жизненный цикл программного средства. Группы процессов жизненного цикла в соответствии со стандартом ISO/IEC 12207.
4. Стадии жизненного цикла ПС, основные этапы.
5. Модели процесса разработки. Водопадный подход (каскадная модель), выводы о применимости классической каскадной модели.
6. Модели процесса разработки. Итерационный подход (спиральная и инкрементальная модели). Гибкие модели процесса разработки.
7. Анализ как начальный этап процесса разработки ПС.
8. Проектирование ПС: концептуальное, логическое, физическое.
9. Реализация (кодирование) и эволюция ПС. Сопровождение ПС.
10. Тестирование и отладка. Основные определения.
11. Виды тестирования.
12. Архитектура ПС. Контроль архитектуры ПС.
13. Унифицированный процесс разработки. Основные цели и отличительные черты RUP.
14. Предмет технологии разработки программного обеспечения согласно RUP.
15. Характеристики качества и критерии качества ПО.
16. Организация разработки ПО. Группа проекта и роли участников группы.
17. Документация, создаваемая в процессе разработки программных средств. Документы управления разработкой ПС. Документы, входящие в состав ПС.
18. Пользовательская документация.
19. Документация по сопровождению программных средств.
20. Объектно-ориентированный подход к разработке ПО. Пакеты классической модели системы.
21. Объектно-ориентированный анализ: назначение, цель и задачи.
22. Объектно-ориентированное проектирование: назначение, цель и задачи.
23. Инструментальные средства проектирования. Системы автоматизации разработки программных систем.

- 24.Международные стандарты проектирования, разработки, оформления документации, пользовательского интерфейса ПИ.
- 25.Сертификация фирм разработчиков по модели качества СММ.
- 26.UML унифицированный язык моделирования. Сущности.
- 27.UML унифицированный язык моделирования. Отношения.
- 28.UML унифицированный язык моделирования. Диаграммы.
- 29.Диаграмма прецедентов. Назначение диаграммы прецедентов. Основные элементы диаграммы.
- 30.Диаграмма прецедентов. Типы отношений на диаграмме прецедентов. Понятие пакета.
- 31.Диаграмма последовательности. Основные типы сообщений используемых для построения диаграммы последовательности. Стереотипы спецификации сообщений.
- 32.Диаграмма классов. Основные понятия (класс, атрибут, квантор видимости, графическое представление класса).
- 33.Диаграмма классов. Типы отношений на диаграмме классов.
- 34.Диаграмма классов. Стереотипы классов (Entity, Control, Boundary).
- 35.Диаграмма кооперации. Назначение диаграммы коопераций. Основные элементы диаграммы. Стереотипы связей.
- 36.Диаграмма состояний. Спецификация состояний (составного состояния, подсостояния, последовательных и параллельных подсостояний).
37. Диаграмма компонентов. Основные понятия (компоненты, типы компонентов, свойства компонентов, стереотипы компонентов).
38. Диаграмма размещения. Назначение диаграммы размещения. Основные элементы диаграммы (узел, процессоры, устройства, связь).

Ст. преподаватель,  О.В. Сташкова