

Документ подписан простой электронной подписью

Информация о владельце: **ФГБОУ ВО «Уральский государственный экономический университет»**

ФИО: Силин Яков Петрович

Должность: Ректор

Дата подписания: 12.05.2023 15:41:29

Уникальный идентификатор документа: **24f866be7aca16484036a8cbb3c509a9531e605f**

Одобрена
Педагогическим советом колледжа

Утверждена

Советом по учебно-методическим
вопросам и качеству образования

протокол № 4 от 14.12.2022 г.

протокол № 4 от 16.12.2022 г.

Директор колледжа  А.Э.Чечулин

(подпись)

Председатель  Д.А. Карх

(подпись)



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины МДК.01.02 Поддержка и тестирование программных модулей

Специальность 09.02.07 Информационные системы и программирование

Форма обучения очная

Год набора 2023

Разработана:
доцент, к.п.н.
Н.С. Кольева

СОДЕРЖАНИЕ

| | |
|--|-----------|
| ВВЕДЕНИЕ | 3 |
| 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ | 3 |
| 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП | 3 |
| 3. ОБЪЕМ ДИСЦИПЛИНЫ | 3 |
| 4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП | 4 |
| 5. ТЕМАТИЧЕСКИЙ ПЛАН | 6 |
| 6. ФОРМЫ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ШКАЛЫ ОЦЕНИВАНИЯ | 7 |
| 7. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ | 9 |
| 8. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ | 13 |
| 9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ | 13 |
| 10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ОНЛАЙН КУРСОВ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ | 14 |
| 11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ | 15 |

ВВЕДЕНИЕ

Рабочая программа дисциплины является частью основной образовательной программы среднего профессионального образования - программы подготовки специалистов среднего звена, разработанной в соответствии с ФГОС СПО

| | |
|----------|---|
| ФГОС СПО | Федеральный государственный образовательный стандарт среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование (приказ Минобрнауки России от 09.12.2016 г. № 1547) |
| ПС | |

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью междисциплинарного курса "Поддержка и тестирование программных модулей" является формирование у обучающихся умений проводить тестирование и отладку программного модуля

Междисциплинарный курс входит в ПМ.01 "Разработка модулей программного обеспечения для компьютерных систем"

В результате освоения междисциплинарного курса обучающийся должен:

Знать:

- основные принципы отладки и тестирования программных продуктов;
- способы оптимизации и приемы рефакторинга;

Уметь:

- выполнять отладку и тестирование программы на уровне модуля;
- уметь выполнять оптимизацию и рефакторинг программного кода;
- оформлять документацию на программные средства.

Иметь практический опыт:

- в использовании инструментальных средств на этапе отладки программного продукта;
- в проведении тестирования программного модуля по определенному сценарию.

Результатом освоения междисциплинарного курса, в соответствии с рабочей программой воспитания, является формирование у обучающихся следующих личностных результатов обучения:

ЛР 10. Бережливо относящийся к природному наследию страны и мира, проявляющий сформированность экологической культуры на основе понимания влияния социальных, экономических и профессионально-производственных процессов на окружающую среду. Выражающий деятельное неприятие действий, приносящих вред природе, распознающий опасности среды обитания, предупреждающий рискованное поведение других граждан, популяризирующий способы сохранения памятников природы страны, региона, территории, поселения, включенный в общественные инициативы, направленные на заботу о них.

ЛР 13. Признающий ценность непрерывного образования, ориентирующийся в изменяющемся рынке труда, избегающий безработицы; управляющий собственным профессиональным развитием; рефлексивно оценивающий собственный жизненный опыт, критерии личной успешности.

ЛР 15. Экономически активный, предприимчивый, готовый к самозанятости.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Дисциплина относится к вариативной части учебного плана.

3. ОБЪЕМ ДИСЦИПЛИНЫ

| Промежуточный контроль | Часов | | | Самостоятельная работа в том числе подготовка контрольных и курсовых |
|------------------------|------------------|--------------------------------|--------------|--|
| | Всего за семестр | Контактная работа (по уч.зан.) | | |
| | | Всего | Лабораторные | |
| | | | | |

| | | | | | |
|-----------|---|----|----|----|---|
| Семестр 5 | | | | | |
| Зачет | 0 | 92 | 92 | 18 | 0 |

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП

В результате освоения ООП у выпускника должны быть сформированы компетенции, установленные в соответствии ФГОС СПО.

Общие компетенции (ОК)

| Шифр и наименование компетенции | Индикаторы достижения компетенций |
|--|---|
| <p>ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;</p> | <p>Уметь:</p> <ul style="list-style-type: none"> - распознавать задачу и/или проблему в профессиональном и/или социальном контексте; - анализировать задачу и/или проблему и выделять ее составные части; - определять этапы решения задачи; - выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы; - составлять план действия; - определять необходимые ресурсы; - владеть актуальными методами работы в профессиональной и смежных сферах; - реализовывать составленный план; оценивать результат и последствия своих действий (самостоятельно или с помощью наставника) <p>Знать:</p> <ul style="list-style-type: none"> - актуальный профессиональный и социальный контекст, в котором приходится работать и жить; - основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте; - алгоритмы выполнения работ в профессиональной и смежных областях; - методы работы в профессиональной и смежных сферах; - структуру плана для решения задач; - порядок оценки результатов решения задач профессиональной деятельности |

| | |
|--|---|
| <p>ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;</p> | <p>Уметь:</p> <ul style="list-style-type: none"> - определять задачи для поиска информации; - определять необходимые источники информации; - планировать процесс поиска; - структурировать получаемую информацию; выделять наиболее значимое в перечне информации; - оценивать практическую значимость результатов поиска; - оформлять результаты поиска; - применять средства информационных технологий для решения профессиональных задач; - использовать современное программное обеспечение; - использовать различные цифровые средства для решения профессиональных задач. <p>Знать:</p> <ul style="list-style-type: none"> - номенклатура информационных источников, применяемых в профессиональной деятельности; - приемы структурирования информации; - формат оформления результатов поиска информации - современные средства и устройства информатизации; - порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств |
| <p>ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.</p> | <p>Уметь:</p> <ul style="list-style-type: none"> - понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы; - участвовать в диалогах на профессиональные темы; - строить простые высказывания о своей профессиональной деятельности; - кратко обосновывать и объяснять свои действия (текущие и планируемые); - писать связные сообщения на профессиональные темы. <p>Знать:</p> <ul style="list-style-type: none"> - лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности; - правила чтения текстов профессиональной направленности |

Профессиональные компетенции (ПК)

| Шифр и наименование компетенции | Индикаторы достижения компетенций |
|---|-----------------------------------|
| Разработка модулей программного обеспечения для компьютерных систем | |

| | |
|---|--|
| <p>ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.</p> | <p>Знать:</p> <ul style="list-style-type: none"> - основные принципы отладки и тестирования программных продуктов; - инструментарий отладки программных продуктов. <p>Уметь</p> <ul style="list-style-type: none"> - выполнять отладку и тестирование программы на уровне модуля; - оформлять документацию на программные средства; - применять инструментальные средства отладки программного обеспечения. <p>Иметь практический опыт:</p> <ul style="list-style-type: none"> - использовать инструментальные средства на этапе отладки программного продукта; - проводить тестирование программного модуля по определенному сценарию. |
| <p>ПК 1.4. Выполнять тестирование программных модулей.</p> | <p>Знать:</p> <ul style="list-style-type: none"> - основные виды и принципы тестирования программных продуктов. <p>Уметь:</p> <ul style="list-style-type: none"> - выполнять отладку и тестирование программы на уровне модуля; - оформлять документацию на программные средства. <p>Иметь практический опыт:</p> <ul style="list-style-type: none"> - проводить тестирование программного модуля по определенному сценарию; - использовать инструментальные средства на этапе тестирования программного продукта. |
| <p>ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.</p> | <p>Знать:</p> <ul style="list-style-type: none"> - способы оптимизации и приемы рефакторинга; - инструментальные средства анализа алгоритма; - методы организации рефакторинга и оптимизации кода; - принципы работы с системой контроля версий. <p>Уметь:</p> <ul style="list-style-type: none"> - выполнять оптимизацию и рефакторинг программного кода; - работать с системой контроля версий. <p>Иметь практический опыт:</p> <ul style="list-style-type: none"> - анализировать алгоритмы, в том числе с применением инструментальных средств; - осуществлять рефакторинг и оптимизацию программного кода. |

5. ТЕМАТИЧЕСКИЙ ПЛАН

| Тема | Часов | | | | | | |
|-----------|--|-------------|--------------------------------|--------------|----------------------|----------------|---------------------------------|
| | Наименование темы | Всего часов | Контактная работа (по уч.зан.) | | | Самост. работа | Контроль самостоятельной работы |
| | | | Лекции | Лабораторные | Практические занятия | | |
| Семестр 5 | | 110 | | | | | |
| Тема 1. | Отладка и тестирование программного обеспечения (ОК 01; ОК 02; ОК 09; ПК 1.3; ПК 1.4; ПК 1.5; ЛР 10; ЛР 13; ЛР 15) | 52 | | 46 | | 6 | |

| | | | | | | | |
|---------|---|----|--|----|--|----|--|
| Тема 2. | Документирование (ОК 01; ОК 02; ОК 09; ПК 1.3; ПК 1.4; ПК 1.5; ЛР 10; ЛР 13; ЛР 15) | 58 | | 46 | | 12 | |
|---------|---|----|--|----|--|----|--|

6. ФОРМЫ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ШКАЛЫ ОЦЕНИВАНИЯ

| Раздел/Тема | Вид оценочного средства | Описание оценочного средства | Критерии оценивания |
|---------------------------------------|-------------------------|---|------------------------------|
| Текущий контроль (Приложение 4) | | | |
| Тема 1 | Тест | Тест состоит из 15 вопросов. Закрытого типа. Количество вариантов - 3 | Оценивается от 2 до 5 баллов |
| Тема 1 | Вопросы | Устный опрос по вопросам. Количество вопросов 13. Количество вариантов - 1. | Оценивается от 2 до 5 баллов |
| Тема 2 | Практическая работа | Работа состоит из 16 вариантов по 1 заданию в каждом варианте. | Оценивается от 2 до 5 баллов |
| Тема 2 | Тест | Тест состоит из 15 вопросов. Закрытого типа. Количество вариантов - 2 | Оценивается от 2 до 5 баллов |
| Промежуточный контроль (Приложение 5) | | | |
| 5 семестр (За) | Билет к зачету | Билет состоит из трех вопросов: 1. Теоретический вопрос. 2. Тестовое задание. 3. Практическое задание. Количество билетов - 25. | Зачет / незачет |

ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

Показатель оценки освоения ООП формируется на основе объединения текущей и промежуточной аттестации обучающегося.

Показатель рейтинга по каждой дисциплине выражается в процентах, который показывает уровень подготовки студента.

Текущая аттестация. Используется 5-балльная система оценивания. Оценка работы студента в течении семестра осуществляется преподавателем в соответствии с разработанной им системой оценки учебных достижений в процессе обучения по данной дисциплине.

В рабочих программах дисциплин (предметов) и практик закреплены виды текущей аттестации, планируемые результаты контрольных мероприятий и критерии оценки учебных достижений.

В течение семестра преподавателем проводится не менее 3-х контрольных мероприятий, по оценке деятельности студента.

Промежуточная аттестация. Используется 5-балльная система оценивания. Оценка работы студента по окончанию дисциплины (части дисциплины) осуществляется преподавателем в соответствии с разработанной им системой оценки достижений студента в процессе обучения по данной дисциплине. Промежуточная аттестация также проводится по окончанию формирования компетенций.

| Показатель оценки | По 5-балльной системе | Характеристика показателя |
|-------------------|-----------------------|--|
| 100% - 85% | отлично | обладают теоретическими знаниями в полном объеме, понимают, самостоятельно умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов на высоком уровне |
| 84% - 70% | хорошо | обладают теоретическими знаниями в полном объеме, понимают, самостоятельно умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов. Могут быть допущены недочеты, исправленные студентом самостоятельно в процессе работы (ответа и т.д.) |
| 69% - 50% | удовлетворительно | обладают общими теоретическими знаниями, умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов на среднем уровне. Допускаются ошибки, которые студент затрудняется исправить самостоятельно. |
| 49 % и менее | неудовлетворительно | обладают не полным объемом общих теоретическими знаниями, не умеют самостоятельно применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов. Не сформированы умения и навыки для решения профессиональных задач |
| 100% - 50% | зачтено | характеристика показателя соответствует «отлично», «хорошо», «удовлетворительно» |
| 49 % и менее | не зачтено | характеристика показателя соответствует «неудовлетворительно» |

7. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

7.2 Содержание практических занятий и лабораторных работ

Тема 1. Отладка и тестирование программного обеспечения (ОК 01; ОК 02; ОК 09; ПК 1.3; ПК 1.4; ПК 1.5; ЛР 10; ЛР 13; ЛР 15)

Лабораторная работа №1 "Методы тестирования"

Выполнение практических заданий по теме

Лабораторная работа №2 "Тестирование на основе потока данных"

Выполнение практических заданий

Лабораторная работа №3 "Тестирование на основе потока управления"

Выполнение практических заданий по теме

Лабораторная работа №4 "Использование инструментальных средств на этапе отладки"

Выполнение практических заданий по теме

Лабораторная работа №5 "Возможности среды разработки для тестирования приложений"

Выполнение практических заданий по теме

Лабораторная работа №6 "Автоматизация тестирования"

Выполнение практических заданий по теме

Лабораторная работа № 7 "Тестирование «белым ящиком»"

Выполнение практических заданий по теме

Лабораторная работа №8 "Тестирование «черным ящиком»"

Выполнение практических заданий по теме

Лабораторная работа №9 "Модульное тестирование"

Выполнение практических заданий по теме

Лабораторная работа №10 "Интеграционное тестирование"

Выполнение практических заданий по теме

Лабораторная работа №11 "Виды ошибок"

Выполнение практических заданий по теме

Лабораторная работа №12 "Методы отладки"

Выполнение практических заданий по теме

Лабораторная работа №13 "Методы тестирования"

Выполнение практических заданий по теме

Лабораторная работа №14 "Классификация тестирования"

Выполнение практических заданий по теме

Лабораторная работа №15 "Классификация тестирования по уровням"

Выполнение практических заданий по теме

Лабораторная работа №16 "Тестирование производительности"

Выполнение практических заданий по теме

Лабораторная работа №17 "Регрессионное тестирование"

Выполнение практических заданий по теме

Лабораторная работа №18 "Тестирование программного модуля по определенному сценарию"

Выполнение практических заданий по теме

Лабораторная работа №19 "Разработка модуля с использованием текстовых компонентов"

Выполнение практических заданий по теме

Лабораторная работа №20 "Отладка и тестирование программы на уровне модуля. Анализ результатов тестирования"

Выполнение практических заданий по теме

Лабораторная работа №21 "Разработка системы тестов на основе потока управления и на основе потока данных"

Выполнение практических заданий по теме

Лабораторная работа №22 "Разработка и отладка модуля вывода и суммирования элементов массива"

Выполнение практических заданий по теме

Лабораторная работа №23 "Разработка и отладка модуля вычисления площади геометрической фигуры"

Выполнение практических заданий по теме

Тема 2. Документирование (ОК 01; ОК 02; ОК 09; ПК 1.3; ПК 1.4; ПК 1.5; ЛР 10; ЛР 13; ЛР 15)

Лабораторная работа №24 "Средства разработки технической документации"

Выполнение практических заданий по теме

Лабораторная работа №25 "Технологии разработки"

Выполнение практических заданий по теме

Лабораторная работа №26 "Документирование программного обеспечения в соответствии с Единой системой программной документации"

Выполнение практических заданий по теме

Лабораторная работа №27 "Автоматизация разработки технической документации"

Выполнение практических заданий по теме

Лабораторная работа №28 "Автоматизированные средства оформления документации"

Выполнение практических заданий по теме

Лабораторная работа №29 "Разработка и отладка модуля сортировки элементов массива"

Выполнение практических заданий по теме

Лабораторная работа №30 "Разработка и отладка модуля шифрования записей текстового файла"

Выполнение практических заданий по теме

Лабораторная работа №31 "Разработка, отладка и оптимизация модуля управления движением объекта по двум координатам"

Выполнение практических заданий по теме

Лабораторная работа №32 "Разработка, отладка и оптимизация модуля отображения элементов двумерного массива"

Выполнение практических заданий по теме

Лабораторная работа №33 "Отладка и оптимизация модуля выполнения операций реляционной алгебры над множествами"

Выполнение практических заданий по теме

Лабораторная работа №34 "Тестирование программного модуля по определенному сценарию"

Выполнение практических заданий по теме

Лабораторная работа №35 "Тестирование программного модуля по определенному сценарию (продолжение)»"

Выполнение практических заданий по теме

Лабораторная работа №36 "Работа с программами и данными в машинном представлении"

Выполнение практических заданий по теме

Лабораторная работа №37 "Работа с программами и данными в машинном представлении"

Выполнение практических заданий по теме

Лабораторная работа №38 "Технология создания ассемблерных программ"

Выполнение практических заданий по теме

Лабораторная работа №39 "Методология разработки технической документации. Технологии разработки документов»"

Выполнение практических заданий по теме

Лабораторная работа №40 "Средства разработки технической документации"

Выполнение практических заданий по теме

Лабораторная работа №41 "Автоматизация разработки технической документации"

Выполнение практических заданий по теме

Лабораторная работа №42 "Автоматизированные средства оформления документации"

Выполнение практических заданий по теме

Лабораторная работа №43 "Оформление документации на программные средства с использованием инструментальных средств (часть 1)"

Выполнение практических заданий по теме

Лабораторная работа №44 "Оформление документации на программные средства с использованием инструментальных средств (часть 2)"

Выполнение практических заданий по теме

Лабораторная работа №45 "Отработка стиля программирования (часть 1)"

Выполнение практических заданий по теме
Лабораторная работа №46 "Работа над индивидуальным проектом" Выполнение практических заданий по теме

7.3. Содержание самостоятельной работы

Тема 1. Отладка и тестирование программного обеспечения (ОК 01; ОК 02; ОК 09; ПК 1.3; ПК 1.4; ПК 1.5; ЛР 10; ЛР 13; ЛР 15)

1. Изучение понятийного аппарата темы, лекционного материала, глав рекомендованных учебников основной и дополнительной литературы.
2. Выполнение заданий для самостоятельной работы
Составить конспект по темам: Виды ошибок. Методы отладки. Методы тестирования
3. Подготовка к лабораторным работам

Тема 2. Документирование (ОК 01; ОК 02; ОК 09; ПК 1.3; ПК 1.4; ПК 1.5; ЛР 10; ЛР 13; ЛР 15)

1. Изучение понятийного аппарата темы, лекционного материала, глав рекомендованных учебников основной и дополнительной литературы.
2. Выполнение заданий для самостоятельной работы
Составить конспект по темам: Средства разработки технической документации. Техно-логии разработки документов
3. Подготовка к лабораторным работам

7.3.1. Примерные вопросы для самостоятельной подготовки к зачету/экзамену
Приложение 1.

7.3.2. Практические задания по дисциплине для самостоятельной подготовки к зачету/экзамену
Приложение 2.

7.3.3. Перечень курсовых работ
Не предусмотрено.

7.4. Электронное портфолио обучающегося
Материалы не размещаются.

7.5. Методические рекомендации по выполнению контрольной работы
Не предусмотрено.

7.6 Методические рекомендации по выполнению курсовой работы
Не предусмотрено.

8. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

По заявлению студента

В целях доступности освоения программы для лиц с ограниченными возможностями здоровья при необходимости кафедра обеспечивает следующие условия:

- особый порядок освоения дисциплины, с учетом состояния их здоровья;
- электронные образовательные ресурсы по дисциплине в формах, адаптированных к ограничениям их здоровья;
- изучение дисциплины по индивидуальному учебному плану (вне зависимости от формы обучения);
- электронное обучение и дистанционные образовательные технологии, которые предусматривают возможности приема-передачи информации в доступных для них формах.
- доступ (удаленный доступ), к современным профессиональным базам данных и информационным справочным системам, состав которых определен РПД.

9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Сайт библиотеки УрГЭУ

<http://lib.usue.ru/>

Основная литература:

1. Лисьев Г.А., Романов П.Ю. Программное обеспечение компьютерных сетей и web-серверов [Электронный ресурс]: Учебное пособие. - Москва: ООО "Научно-издательский центр ИНФРА-М", 2021. - 145 – Режим доступа: <https://znanium.com/catalog/product/1189343>
2. Исаченко О.В. Программное обеспечение компьютерных сетей [Электронный ресурс]: Учебное пособие. - Москва: ООО "Научно-издательский центр ИНФРА-М", 2021. - 158 – Режим доступа: <https://znanium.com/catalog/product/1189344>
3. Гниденко И. Г., Павлов Ф. Ф. Технология разработки программного обеспечения [Электронный ресурс]: Учебное пособие Для СПО. - Москва: Юрайт, 2021. - 235 – Режим доступа: <https://urait.ru/bcode/472502>
4. Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности [Электронный ресурс]: Учебное пособие. - Москва: ООО "КУРС", 2022. - 336 – Режим доступа: <https://znanium.com/catalog/product/1858587>
5. Лисьев Г.А., Романов П.Ю., Аскерко Ю.И. Программное обеспечение компьютерных сетей и web-серверов [Электронный ресурс]: Учебное пособие. - Москва: ООО "Научно-издательский центр ИНФРА-М", 2023. - 145 с. – Режим доступа: <https://znanium.com/catalog/product/1878635>

Дополнительная литература:

1. Партыка Т. Л., Попов И.И. Операционные системы, среды и оболочки [Электронный ресурс]: Учебное пособие. - Москва: Издательство "ФОРУМ", 2021. - 560 с. – Режим доступа: <https://znanium.com/catalog/product/1189335>
2. Кузин А. В., Чумакова Е. В. Программирование на языке Си [Электронный ресурс]: Справочник: Учебное пособие. - Москва: Издательство "ФОРУМ", 2023. - 143 с. – Режим доступа: <https://znanium.com/catalog/product/1878382>
3. Федоров Д. Ю. Программирование на языке высокого уровня Python [Электронный ресурс]: учебное пособие для спо. - Москва: Юрайт, 2023. - 214 с – Режим доступа: <https://urait.ru/bcode/510042>
4. Казанский А. А. Программирование на Visual C# [Электронный ресурс]: учебное пособие для спо. - Москва: Юрайт, 2023. - 192 с – Режим доступа: <https://urait.ru/bcode/513400>
5. Кудрявцева И., Швецкий М. В. Программирование: комбинаторная логика [Электронный ресурс]: учебное пособие для спо. - Москва: Юрайт, 2023. - 524 с – Режим доступа: <https://urait.ru/bcode/517677>

6. Байбородова Л. В., Куприянова Г. В., Степанов Е. Н., Золотарева А. В., Кораблева А. А., Кириченко Е. Б., Паладьев С. Л., Харисова И. Г., Чернявская А. П. Педагогические технологии в 3 ч. Часть 3. Проектирование и программирование [Электронный ресурс]: учебник и практикум для вузов. - Москва: Юрайт, 2023. - 219 с – Режим доступа: <https://urait.ru/bcode/516050>

10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ОНЛАЙН КУРСОВ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Перечень лицензионного программного обеспечения:

Microsoft Windows 10 .Договор № 52/223-ПО/2020 от 13.04.2020, Акт № Tr000523459 от 14.10.2020. Срок действия лицензии 30.09.2023.

Astra Linux Common Edition. Договор № 1 от 13 июня 2018, акт от 17 декабря 2018. Срок действия лицензии - без ограничения срока.

Microsoft Office 2016. Договор № 52/223-ПО/2020 от 13.04.2020, Акт № Tr000523459 от 14.10.2020 Срок действия лицензии 30.09.2023.

МойОфис стандартный. Соглашение № СК-281 от 7 июня 2017. Дата заключения - 07.06.2017. Срок действия лицензии - без ограничения срока.

Libre Office. Лицензия GNU LGPL. Срок действия лицензии - без ограничения срока.

Adobe Acrobat DC Pro. Договор № 158/223-ПО/2022 от 15.12.2022. Срок действия лицензии 15.12.2023.

Adobe InCopy CC. Договор № 158/223-ПО/2022 от 15.12.2022. Срок действия лицензии 15.12.2023.

Adobe Illustrator CC. Договор № 158/223-ПО/2022 от 15.12.2022. Срок действия лицензии 15.12.2023.

Adobe Premiere Pro CC. Договор № 158/223-ПО/2022 от 15.12.2022. Срок действия лицензии 15.12.2023.

GIMP. Лицензия GNU GENERAL PUBLIC LICENSE. Срок действия лицензии - без ограничения срока.

Inkscape. Лицензия GNU GENERAL PUBLIC LICENSE. Срок действия лицензии - без ограничения срока.

Maple 11. Договор № 67Т от 04.07.2007 г..

InterBase XE7 ToGo. Эл. лицензия, Информационное письмо.

Microsoft Dynamics CRM. Соглашение от 23.08.2016.

Microsoft Visual Studio Community. Лицензия для образовательных учреждений. Срок действия лицензии - без ограничения срока.

Microsoft SQL Server Express. Лицензия для образовательных учреждений. Срок действия лицензии - без ограничения срока.

MySQL Community Server. Стандартная общественная лицензия GNU (GPL). Срок действия лицензии - без ограничения срока.

PostgreSQL Server. Лицензия PostgreSQL. Срок действия лицензии - без ограничения срока.

Язык программирования R. Лицензия GNU GPL 2. Срок действия лицензии - без ограничения срока.

R Studio (среда для языка программирования R). Лицензия GNU Affero General Public License v3. Срок действия лицензии - без ограничения срока.

Язык программирования Python. Python Software Foundation License (PSFL). Срок действия лицензии - без ограничения срока.

Putty. Лицензия MIT license. Срок действия лицензии - без ограничения срока.

Архиватор 7-Zip. Лицензия GNU LGPLv2.1 + with unRAR restriction / LZMA SDK in the public domain. Срок действия лицензии - без ограничения срока.

FAR Manager. Лицензия Revised BSD license. Срок действия лицензии - без ограничения срока.

Notepad++. Лицензия GNU General Public License. Срок действия лицензии - без ограничения срока.

Язык программирования Java.

Перечень информационных справочных систем, ресурсов информационно-телекоммуникационной сети «Интернет»:

11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Реализация учебной дисциплины осуществляется с использованием материально-технической базы УрГЭУ, обеспечивающей проведение всех видов учебных занятий и научно-исследовательской и самостоятельной работы обучающихся:

Специальные помещения представляют собой учебные аудитории для проведения всех видов занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду УрГЭУ.

Все помещения укомплектованы специализированной мебелью и оснащены мультимедийным оборудованием спецоборудованием (информационно-телекоммуникационным, иным компьютерным), доступом к информационно-поисковым, справочно-правовым системам, электронным библиотечным системам, базам данных действующего законодательства, иным информационным ресурсам служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа презентации и другие учебно-наглядные пособия, обеспечивающие тематические иллюстрации.

7.3.1. Примерные вопросы для самостоятельной подготовки к зачету

1. Введение: тестирование - способ обеспечения качества программного продукта
2. Основные понятия тестирования
3. Подходы к обоснованию истинности формул и программ и их связь с тестированием.
4. Вопросы организации тестирования.
5. Фазы тестирования, основные проблемы тестирования и поставлена задача выбора конечного набора тестов.
6. Требования к идеальному критерию тестирования и классы частных критериев.
7. Особенности применения структурных и функциональных критериев на базе конкретных примеров.
8. Особенности применения методов стохастического тестирования и метод оценки скорости выявления ошибок.
9. Мутационный критерий и на примере иллюстрируется техника работы с ним.
10. Оценка оттестированности проекта: метрики и методика интегральной оценки
11. Графовые модели проекта, метрики оценки оттестированности проекта, приводятся примеры плоской и иерархической моделей проекта.
12. Особенности модульного тестирования, подходы к тестированию на основе потока управления, потока данных, динамические и статические методы при структурном подходе. Взаимосвязь сборки модулей и методов интеграционного тестирования.
13. Подходы монолитного, инкрементального, нисходящего и восходящего тестирования. Рассматриваются особенности интеграционного тестирования в процедурном программировании.
14. Интеграционное тестирование и его особенности для объектно-ориентированного программирования
15. Модель объектно-ориентированной программы, использующая понятие Р-путей и ММ-путей.
16. Оценки сложности тестирования и методика тестирования объектно-ориентированной программы. Рассматривается пример интеграционного тестирования.
17. Разновидности тестирования: системное и регрессионное тестирование
18. Автоматизация тестирования
19. Особенности индустриального тестирования
20. Документирование и оценка индустриального тестирования. Описываются особенности документирования тестовых процедур для ручных и автоматизированных тестов, описаний тестовых наборов и тестовых отчетов. Рассматривается жизненный цикл дефекта. Обсуждаются метрики, используемые при тестировании.
21. Регрессионное тестирование: цели и задачи, условия применения, классификация тестов и методов отбора
22. Регрессионное тестирование: разновидности метода отбора тестов
23. Регрессионное тестирование: методики, не связанные с отбором тестов и методики порождения тестов
24. Регрессионное тестирование: алгоритм и программная система поддержки Описание тестируемой системы и ее окружения. Планирование тестирования
25. Модульное тестирование на примере классов
26. Интеграционное тестирование
27. Системное тестирование
28. Ручное тестирование
29. Автоматизация тестирования с помощью скриптов
30. Автоматическая генерация тестов на основе формального описания
31. Описание ручного тестирования
32. Автоматизация тестирования с помощью скриптов
33. Описание автоматической генерации MSC тестов

7.3.2. Практические задания по междисциплинарному курсу для самостоятельной подготовки к зачету

ОК 01.: Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

Закрытые вопросы.

1. Выберите один правильный ответ из предложенных вариантов.

В основные обязанности тестировщика входят:

- a. **Выявление ошибки**
- b. Исправление ошибки
- c. Объяснение причины ошибки

2. Выберите один правильный ответ из предложенных вариантов.

Одному тест-требованию может соответствовать:

1. только один тестовый пример
2. **несколько тестовых примеров**
3. не более двух тестовых примеров

3. Выберите один правильный ответ из предложенных вариантов.

Два тестовых примера проверяют один и тот же класс эквивалентности:

- a. если от них получен один и тот же результат
- b. **если от них получена одинаковая реакция системы**
- c. если они построены по одному тест-требованию

4. Выберите один правильный ответ из предложенных вариантов.

Ручное тестирование целесообразно применять:

- a. **если тестовый пример не может быть выполнен в автоматическом режиме**
- b. если тестовый пример построен по одному тест-требованию
- c. если автоматизация выполнения тестового примера очень проста

5. Выберите один правильный ответ из предложенных вариантов.

Тестовое окружение может использоваться для:

- a. **запуска и выполнения тестируемого модуля**
- b. сбора ожидаемых выходных данных
- c. сравнения реальных входных данных

Открытые вопросы.

1. Сколько классов эквивалентности в общем случае выделяют для функции с двумя целочисленными входными параметрами и одним целочисленным выходным значением?

Ответ: 4

2. Составьте соответствие

| Имя файла | Примечание |
|-------------------|--|
| AuthoringTest.txt | Примечания о создании тестов, включающие инструкции по добавлению дополнительных тестов к проекту |
| CalcClassTest.cs | Включает в себя сгенерированный тест для тестирования метода Add () наряду с методами для тестовой инициализации и очистки |
| ManualTest1.mht | Шаблон, который заполняется инструкциями при ручном тестировании |
| UnitTest1.cs | Пустая структура unit test класса, куда помещаются дополнительные тесты |

Ответ: 1-1; 2-2; 3-3; 4-4

3. Выберите верные утверждения:

1. Полное покрытие по веткам дает полное покрытие по строкам.
2. Полное покрытие по веткам не дает полного покрытия по строкам.
3. Полное покрытие по строкам без ветвления дает полное покрытие кода по веткам.
4. Полное покрытие по MC\DC не дает полного покрытия по строкам.

Ответ: 1,3

4. Какие условия должны быть выполнены для обеспечения полного покрытия по методу MC\DC?

1. должно быть показано зависимое влияние каждой из компонент на значение логического условия
2. каждое логическое условие должно принимать все возможные значения
3. каждая компонента логического условия должна хотя бы один раз принимать все возможные значения
4. любая часть логического условия должна принимать хотя бы раз все возможные значения
5. должно быть показано независимое влияние каждой из компонент на значение логического условия

Ответ: 2, 3, 5

5. Для каких видов интеграционного тестирования нужны заглушки?

1. восходящего
2. монолитного
3. нисходящего
4. с поздней интеграцией
5. с постоянной интеграцией
6. с регулярной интеграцией

Ответ: 1, 3, 6

ОК 02.: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;

Закрытые вопросы.

1. Выберите один правильный ответ из предложенных вариантов.

Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

2. Выберите один правильный ответ из предложенных вариантов.

Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований

2. **программный модуль, т.е. минимальный компилируемый элемент программной системы**

3. задача в списке задач проекта

3. Выберите один правильный ответ из предложенных вариантов.

Какие основные задачи решаются в ходе модульного тестирования?

1. **Поиск и документирование несоответствий требованиям**

2. Рефакторинг модулей

3. Отладка

4. Выберите один правильный ответ из предложенных вариантов.

Тестовое окружение для программного кода на структурных языках программирования состоит из:

1. **драйвера**

2. тестов

3. исходного кода

5. Выберите один правильный ответ из предложенных вариантов.

Полная система тестов позволяет утверждать, что:

1. **система реализует всю функциональность, указанную в требованиях**

2. система работает корректно

3. система работает правильно

Открытые вопросы.

1. Процесс обнаружения и исправления ошибок называют ...

Ответ: Отладкой

2. Допишите предложение. В бизнес-процессах выделяют классы процессов:

Ответ: Основные бизнес-процессы

3. Выберите верный ответ. Вариантов несколько. К малым интегрированным средствам моделирования относятся:

1. ARIS Toolset

2. Design/IDEF

3. ERwin

4. BPwin

5. Designer/2000

6. Paradigm Plus

7. Model Mart

8. Rational Rose

Ответ: ERwin, BPwin, Model Mart

4. Когда система передана заказчику, начинается этап ...

Ответ: Эксплуатации

5. Для каких видов интеграционного тестирования нужен драйвер?

1. восходящего

2. монолитного

3. нисходящего

4. с поздней интеграцией
 5. с постоянной интеграцией
 6. с регулярной интеграцией
- Ответ: 1, 5, 6

ОК 09.: Пользоваться профессиональной документацией на государственном и иностранном языках.

Закрытые вопросы.

1. Выберите один правильный ответ из предложенных вариантов.

Согласно методу МС\ДС для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. **4-х тестовых примеров**
3. 5-х тестовых примеров

2. Выберите один правильный ответ из предложенных вариантов.

Одной из основных задач анализа полноты покрытия кода является:

1. выявление участков кода, которые выполняются при выполнении тестовых примеров
2. **выявление участков кода, которые содержат ошибки**
3. выявление участков кода, которые не выполняются при выполнении тестовых примеров

3. Выберите один правильный ответ из предложенных вариантов.

При использовании какого метода интеграционного тестирования сначала все программные модули, входящие в состав системы, тестируются и только затем объединяются для интеграционного тестирования?

1. **восходящего**
2. монолитного
3. нисходящего

4. Выберите один правильный ответ из предложенных вариантов.

При использовании какого метода интеграционного тестирования подразумевается, что, как только разрабатывается новый модуль системы, он сразу же интегрируется со всей остальной системой?

1. восходящего
2. монолитного
3. **с постоянной интеграцией**

5. Выберите один правильный ответ из предложенных вариантов.

Описательные спецификации могут быть ...

1. последовательными
2. параллельными
3. **алгебраическими**

Открытые вопросы.

1. К средним интегрированным средствам моделирования относятся:

1. Rational Rose
2. Design/IDEF
3. BPwin

4. Designer/2000
5. ARIS Toolset
6. Model Mart
7. Paradigm Plus
8. ERwin

Ответ: Design/IDEF, Designer/2000, ARIS Toolset

2. Объектно-ориентированная методология (ООМ) включает в себя составные части:

1. Объектно-ориентированный анализ
2. Объектно-ориентированный подкласс
3. Объектно-ориентированное проектирование
4. Объектно-ориентированная парадигма
5. Объектно-ориентированная экспозиция
6. Объектно-ориентированное моделирование
7. Объектно-ориентированное программирование
8. Объектно-ориентированная декомпозиция

Ответ: 1, 3, 7

3. Прием инженерии программного обеспечения – это ...

Ответ: техническая реализация проекта командой

4. Какие этапы включает в себя каскадная модель разработки программного обеспечения?

Ответ: анализ, планирование, кодирование и внедрение

5. Для каких видов интеграционного тестирования при разработке часто выполняется интеграция?

1. восходящего
2. монолитного
3. нисходящего
4. с поздней интеграцией
5. с постоянной интеграцией
6. с регулярной интеграцией

Ответ: 3, 5, 6

ПК 1.3.: Выполнять отладку программных модулей с использованием специализированных программных средств.

Закрытые вопросы.

1. Выберите один правильный ответ из предложенных вариантов.

При тестировании методом черного ящика используются следующие критерии:

1. графа причин и следствий

2. покрытия операторов
3. покрытия ребер

2. Выберите один правильный ответ из предложенных вариантов.

Предусмотрение изменений – это принцип, который влияет на такие качества программного обеспечения как ...

1. повторную применимость

2. прозрачность

3. детерминированность реализации

3. Выберите один правильный ответ из предложенных вариантов.

Методы и технологии реинжиниринга и обратного инжиниринга программного обеспечения нацелены на ...

1. тестирование унаследованного программного обеспечения
- 2. оптимизацию унаследованного программного обеспечения**
3. модификацию унаследованного программного обеспечения

4. Выберите один правильный ответ из предложенных вариантов.

С точки зрения разработчика программного обеспечения качество последнего заключается в ...

1. производительности
2. легкости применения
- 3. переносимости**

5. Выберите один правильный ответ из предложенных вариантов.

Верифицируемость программного продукта предполагает ...

- 1. достаточность тестирования свойств системы**
2. единообразие пользовательского интерфейса
3. формальное описание устойчивости

Открытые вопросы.

1. Если дефекты программного обеспечения могут быть устранены применяемыми усилиями, то о таком программном обеспечении говорят как о ...

Ответ: Ремонтпригодном

2. Установите соответствие

| | |
|-------------|---|
| 1. Объект | а) Атрибуты (основные характеристики), которые описывают особенности объекта (цвет, ширина, положение и т.д.) |
| 2. Событие | б) Совокупность данных (компонентов) и методов работы с ними |
| 3. Свойство | с) Отклик на внешнее воздействие |

Ответ: 1 – б, 2 – с, 3 - а

3. Программное сопровождение подразделяют на три категории:

Корректирующее
изменяющее
Совершенствующее
Настраивающее
формирующее

Ответ: 1, 3, 4

4. Установите соответствие

| | |
|---------------------------|--|
| 1. Application.Initialize | 1. загружает и инициализирует форму |
| 2. Application.CreateForm | 2. активизирует форму и начинает выполнение приложения |
| 3. Application.Run | 3. подготавливает приложение к работе |

Ответ: 1 – с, 2 – а, 3 - б

5. Дополнить предложение. Основным окном разрабатываемого приложения является ...

Ответ: Форма

ПК 1.4.: Выполнять тестирование программных модулей.

Закрытые вопросы

1. Выберите один правильный ответ из предложенных вариантов.

Программная инженерия:

1. **software engineering**
2. Инструменты создания программного обеспечения
3. Коллектив инженеров-программистов, разрабатывающих программное обеспечение для компьютеров

2. Выберите один правильный ответ из предложенных вариантов.

Построение SADT-модели включает в себя выполнение следующих действий:

1. Написание программного обеспечения для разрабатываемой системы по требованиям заказчика
2. **Сбор информации об объекте, определение его границ**
3. Представление исследуемой системы в графическом виде

3. Выберите один правильный ответ из предложенных вариантов.

Моделирование основывается на принципах:

1. **Выбор модели оказывает определяющее влияние на подход к решению проблемы и на то, как будет выглядеть это решение**
2. Декомпозиции системы на отдельные подзадачи
3. Инкапсуляции и полиморфизма

4. Выберите один правильный ответ из предложенных вариантов.

CASE-средства классифицируются по следующим признакам:

1. По применяемым методологиям и моделям систем и БД
2. По используемому программному обеспечению
3. По этапам жизненного цикла программного обеспечения

5. Выберите один правильный ответ из предложенных вариантов.

К основным понятиям объектно-ориентированного подхода относятся:

1. Полиморфизм

2. Реализация
3. Агрегирование

Открытые вопросы.

1. Установите соответствие

| | |
|----------------|--|
| 1. Standart | 1. Обеспечивает доступ к 32-битным элементам Windows |
| 2. Additional | 2. Компоненты, реализующие интерфейс с пользователем и процесс управления данными для БД |
| 3. Win32 | 3. Включает стандартные компоненты, обеспечивающие некоторые функции интерфейса пользователя |
| 4. System | 4. Специализированные компоненты, организующие доступ к БД |
| 5. DataAccess | 5. Набор компонентов для доступа к системным ресурсам (OLE, DDE) |
| 6. DataControl | 6. Дополнительные интерфейсные компоненты для красочного оформления приложения |

Ответ: 1 – с, 2 – f, 3 – а, 4 - е, 5 – d, 6 – b

2. Установите соответствие

| | |
|------------------------------|--|
| 1. Файл проекта | 1. (.pas) соответствующий файл модуль для хранения кода |
| 2. Файл модуля | 2. (.res) содержит пиктограмму и прочие ресурсы |
| 3. Файл формы | 3. (.dpr) текстовый файл используется для хранения информации о формах и модулях, содержит операторы инициализации и запуска программы на выполнение |
| 4. Файл опций проекта | 4. (.dfm) двоичный файл, который создается для хранения информации о ваших формах и фреймах |
| 5. Файл ресурсов | 5. (.cfg) хранит установки проекта |
| 6. Файл конфигурации проекта | 6. (.dof) хранит установки опций проекта |

Ответ: 1 – с, 2 – а, 3 – d, 4 - f, 5 – b, 6 – e

3. Главные принципы объектного подхода:

1. Абстрагирование
2. Наследование
3. Ограничение доступа или инкапсуляция
4. Безграничный доступ или инкапсуляция
5. Модульность и иерархия

Ответ: Абстрагирование, Ограничение доступа или инкапсуляция, Модульность и иерархия

4. Выберите дополнительные принципы объектного подхода:

1. Реализация
2. Типизация
3. Параллелизм
4. Внедрение
5. Перпендикулярность
6. Сохраняемость или устойчивость

Ответ: Типизация, Параллелизм, Сохраняемость или устойчивость

5. К инструментальным средствам объектно-ориентированного анализа и проектирования относятся:

1. Rational Rose
2. Model Mart
3. MS Visio
4. ARIS
5. IDEF1X
6. Erwin
7. BPwin
8. JAM

Ответ: Rational Rose, MS Visio, ARIS

ПК 1.5.: Осуществлять рефакторинг и оптимизацию программного кода.

Закрытые вопросы

1. Выберите один правильный ответ из предложенных вариантов.

К инструментальным средствам представления функциональных моделей относятся:

1. JAM
2. **Model Mart**
3. MS Visio

2. Выберите один правильный ответ из предложенных вариантов.

Методологии, поддерживаемые в BPwin:

1. IDEF1X
2. **IDEF0**

3. IDEF1

3. Выберите один правильный ответ из предложенных вариантов.

Уровни логической модели:

1. Диаграмма сущность
2. Диаграмма пакетов
3. Диаграмма сущность-связь

4. Выберите один правильный ответ из предложенных вариантов.

Типы стрелок не входящие в состав диаграммы IDEF0:

1. Input
2. Editor
3. Control

5. Выберите один правильный ответ из предложенных вариантов.

Поддерживаемые в RPTwin типы операторов:

1. Текстовый оператор конкатенации (&)
2. Символ
3. Текст

Открытые вопросы.

1. Диаграмма IDEF0 может содержать следующие типы диаграмм:

1. Диаграмму классов
2. + Контекстную диаграмму, диаграмму декомпозиции
3. Диаграмму компонентов
- 4.+ Диаграмму дерева узлов
5. Диаграмму взаимодействий
- 6.+ Диаграмму только для экспозиции (FEO)
7. Диаграмму последовательности, диаграмму кооперации
8. Диаграмму узлов

Ответ: 2, 4, 6

2. Дополните предложение .Внутренние стрелки не входящие в состав диаграммы IDEF0:

Ответ: mechanism- output, mechanism- input

3. Дополните предложению BPwin допускает следующие переходы с одной нотации на другую:

Ответ: DFD, IDEF3, IDEF3

4. Дополнить предложение. DFD описывает:

Ответ: Функции обработки информации (работы)

5. Дополнить предложение. BPwin позволяет создавать на диаграмме DFD типы граничных стрелок:

Ответ: Обычная граничная стрелка

Практические задания

1. Разработать программу, которая принимает от пользователя квадратную матрицу, вычисляет сумму элементов, лежащих выше главной диагонали. Предусмотреть пользовательский интерфейс для программы.

2. Разработать программу, которая принимает от пользователя квадратную матрицу, вычисляет сумму элементов, лежащих выше побочной диагонали. Разработать пользовательский интерфейс для программы.

3. Разработать программу, позволяющую переводить двоичные целые числа в десятичные и, наоборот, десятичные – в двоичные. Предусмотреть пользовательский интерфейс.

4. Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет сумму отрицательных элементов массива, произведение элементов массива, расположенных между максимальным и минимальным элементами. Предусмотреть пользовательский интерфейс для программы.

5. Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет сумму положительных элементов массива, произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами. Предусмотреть пользовательский интерфейс для программы.

6. Разработать программу, которая принимает от пользователя массив, состоящий из n целых элементов, преобразовывает массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях. Предусмотреть пользовательский интерфейс для программы.

7. Разработать программу, которая принимает от пользователя массив, состоящий из n целых элементов, преобразовывает массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных. Предусмотреть пользовательский интерфейс для программы.

8. Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет количество элементов массива, больших C , произведение элементов массива, расположенных после максимальным по модулю элемента. Предусмотреть пользовательский интерфейс для программы.

9. Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, упорядочивает по возрастанию отдельно элементы, стоящие на четных местах, и элементы, стоящие на нечетных местах. Предусмотреть пользовательский интерфейс для программы.

10. Разработать программу, которая принимает от пользователя целочисленную квадратную матрицу, определяет сумму элементов в тех строках, которые не содержат отрицательных элементов. Предусмотреть пользовательский интерфейс для программы.

11. Разработать программу, которая определяет, сколько раз встретилось заданное слово в текстовом файле, длина строки в котором не превышает 80 символов. Текст не содержит переносов слов. Предусмотреть пользовательский интерфейс для программы.

12. Разработать программу, которая считывает текст из файла и выводит на экран только вопросительные предложения из этого текста. Предусмотреть пользовательский интерфейс для программы.

13. Разработать программу, которая считывает текст из файла и выводит на экран только предложения, не содержащие запятых. Предусмотреть пользовательский интерфейс для программы.

14. Разработать программу, которая считывает текст из файла и выводит на экран предложения, содержащие максимальное количество знаков пунктуации. Предусмотреть пользовательский интерфейс для программы.
15. Разработать программу, которая считывает текст из файла и выводит на экран сначала вопросительные, а затем восклицательные предложения. Предусмотреть пользовательский интерфейс для программы.
16. Разработать программу, которая вычисляет и выводит на экран и в файл в виде таблицы значения функции F на интервале от -2 до 4 с шагом $0,25$.
17. Разработать программу, которая вычисляет и выводит на экран и в файл в виде таблицы значения функции F на интервале от -5 до 5 с шагом $0,5$.
18. Разработать программу, которая вычисляет и выводит на экран и в файл в виде таблицы значения функции F на интервале от -3 до 3 с шагом $0,5$.
19. Разработать программу, которая вычисляет и выводит на экран и в файл в виде таблицы значения функции F на интервале от -3 до 5 с шагом $0,3$.
20. Разработать программу, которая вычисляет и выводит на экран и в файл в виде таблицы значения функции F на интервале от -1 до 4 с шагом $0,2$.
21. Разработать программу, которая для вещественного массива из n элементов определяет сумму его элементов, расположенных правее последнего отрицательного элемента. Результат записать в файл. Предусмотреть пользовательский интерфейс для программы.
22. Разработать программу, которая для целочисленной матрицы определяет среднее арифметическое ее элементов и количество положительных элементов в каждой строке. Результат записать в файл. Предусмотреть пользовательский интерфейс для программы.
23. Разработать программу, которая определяет, встречается ли в заданном текстовом файле заданная последовательность символов. Длина строки текста не превышает 80 символов, текст не содержит переносов слов, последовательность не содержит пробельных символов.
24. Разработать программу записи в файл последовательности чисел натурального ряда до $n=100$.
25. Разработать программу, которая считывает текст из файла и выводит на экран только вопросительные предложения из этого текста.
26. Разработать программу, которая считывает текст из файла и выводит на экран только восклицательные предложения из этого текста.
27. Разработать программу, которая считывает текст из файла и выводит на экран только предложения, заключенные в кавычки.
28. Разработать программу, которая считывает текст из файла и определяет, сколько в нем слов, состоящих из не более чем 4 букв.
29. Разработать программу, которая считывает текст из файла и выводит на экран количество слов, начинающихся с гласной буквы.
30. Разработать программу, которая считывает текст из файла и выводит на экран самое длинное предложение.

**Приложение 4
к рабочей программе**

Федеральное государственное бюджетное образовательное учреждение высшего образования
УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНЫ
на заседании Педагогического совета колледжа

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ
ТЕКУЩЕГО КОНТРОЛЯ**

междисциплинарный курс

Поддержка и тестирование программных модулей

1-вариант

1. Delphi – это...
 - 1) Язык программирования низкого уровня
 - 2) **Система объектно-ориентированного визуального программирования**
 - 3) Программа обработки видео
 - 4) Программа для перевода кода программы на язык машинных кодов.
2. RAD – это...
 - 1) Программа записи видео
 - 2) Средство модульного программирования
 - 3) **Среда быстрой разработки приложений**
3. Язык программирования Delphi
 - 1) **Object Pascal**
 - 2) Java
 - 3) C#
4. IDE – это...
 - 1) История развития Delphi
 - 2) Приложение, разрабатываемое программистом
 - 3) **Интегрированная среда разработки**
5. IDE не включает в себя: *(несколько вариантов ответа)*
 - 1) Редактор кода
 - 2) Высокопроизводительный компилятор в машинный код
 - 3) Объектно-ориентированную модель компонент
 - 4) **Эффекты и переходы**
 - 5) Визуальное построение приложений
 - 6) **Сопровождение ПП**
 - 7) Средство для построения баз данных
6. Объектно-ориентированная программа - ...
 - 1) Программирование, основанное на объектах
 - 2) **Совокупность объектов и способов их взаимодействия**
 - 3) Структура среды программирования
7. Дополнить предложение. Основным окном разрабатываемого приложения является ...
 - 1) Код
 - 2) **Форма**
 - 3) Библиотека
 - 4) Объект
8. Дополнить предложение. Коды для стандартных окон диалога и кнопок в системе Delphi получены от ...
 - 1) Компилятора
 - 2) C++
 - 3) **Windows**
9. Названия процедур на определенное событие:
 - 1) Компоненты
 - 2) **Обработчики событий**
 - 3) Редактор кода
10. Назначение главного меню:
 - 1) **Осуществление функций управления при разработке программ**
 - 2) Сопровождение программных продуктов
 - 3) Автоматизирует процесс тестирования программ
11. Назначение панели инструментов:

- 1) Обработка событий
 - 2) **Меню команд быстрого доступа к командам, содержащее набор кнопок, функции которых эквивалентны наиболее часто употребляемым командам Главного меню**
 - 3) Построение баз данных
12. Назначение палитры компонентов:
- 1) **содержит пиктограммы, которые представляют компоненты VCL , которые можно включить в приложение**
 - 2) Подделка подписей
 - 3) Управление Paint
13. Дополнить предложение. Главное окно управляет окнами ... *(несколько вариантов ответа)*
- 1) **Инспектор объектов**
 - 2) Язык ассемблера
 - 3) **Редактора кода**
 - 4) **Проектировщика форм**
 - 5) Трансляции программы
14. Заготовка разрабатываемого приложения - ...
- 1) Компилятор
 - 2) Свойство
 - 3) **Окно проектировщика форм**
 - 4) Главное окно
15. Действия которые нельзя выполнить с помощью проектировщика форм
- 1) Добавить компоненты в форму
 - 2) **Сменить язык программирования**
 - 3) Модифицировать форму и её компоненты
 - 4) Связать обработчики событий компонента с программой на Object Pascal, содержащейся в редакторе кода
 - 5) **Документирование программы**

Вариант -2

1. Передача фокуса ввода на редактор ввода
 - a. Label1.SetFocus;
 - b. Edit1.Add;
 - c. **Edit1.SetFocus;**
2. Открыть файл проекта Project1 можно, нажав:
 - a. CTRL+F4
 - b. CTRL+ALT+F10
 - c. **CTRL+F12**
 - d. ALT+F12
3. Для обозначения комментария не используются:
 - a. (*комментарий*)
 - b. **\комментарий/**
 - c. {комментарий}
 - d. //комментарий
4. Зарезервированное слово, объявляющее блок подключаемых к проекту модулей:
 - a. Program
 - b. Begin..end
 - c. **Uses**
 - d. Forms
5. Дополнить предложение. Объект Application определяет ...
 - a. **Свойства и методы программы-приложения для Windows**

- b. Отображение размещения формы
 - c. Перемещение по программному коду
6. Метод Application.Run:
- a. Создает окно формы, регистрирует входящие в него компоненты
 - b. Подключает автоматический цикл обработки сообщений Windows к приложению**
 - c. Выполняет подпрограммы раздела инициализации модулей приложения в случае их наличия
7. Контейнер, содержащий все элементы программы:
- a. Unit1
 - b. Form1**
 - c. Edit1
 - d. Memo1
8. Дополнить предложение. Свойство **Position** определяет ...
- a. Размещение и размер формы**
 - b. Подключение модулей
 - c. Цвет шрифта
9. **poDesigned:**
- a. Форма выводится в центр экрана
 - b. Windows автоматически определяет начальную позицию и размеры формы
 - c. Форма отображается в той позиции и с теми размерами, которые были установлены при её конструировании**
10. Определяет способ выравнивания компонента внутри контейнера:
- a. Свойство Position
 - b. Свойство Align**
 - c. Свойство Memo
 - d. Свойство ScrollBars
11. **alClient:**
- a. Компонент занимает всю поверхность контейнера**
 - b. Выравнивание не используется
 - c. Компонент помещается в нижнюю часть контейнера
12. **alCustom:**
- a. Компонент помещается в верхнюю часть контейнера
 - b. Компонент помещается в нижнюю часть контейнера
 - c. Размеры и положение компонента устанавливаются разработчиком**
13. Объект **Memo1** - ...
- a. Однострочный редактор
 - b. Многострочный редактор, содержащий несколько строк текста**
 - c. Текстовая надпись
14. Свойство **ScrollBars:**
- a. Задаёт в поле редактирования полосы прокрутки**
 - b. Определяет способ выравнивания компонента внутри контейнера
 - c. Задаёт имя объекта
15. Функциональная кнопка с изображением устанавливается из:
- a. Палитра компонентов -> Standart -> Button
 - b. Палитра компонентов -> Standart -> Label
 - c. Палитра компонентов -> Additional -> BitBtn**

3- вариант

1. Дополнить предложение. Свойства Height и Width показывают...
 - a. Какое действие должен выполнить программист
 - b. Размеры компонента**
 - c. Исполняемый блок программы
2. Дополнить предложение. Проект Delphi представляет собой...
 - a. Набор программных единиц – модулей**
 - b. Основной интерфейсный элемент в Delphi
 - c. Атрибуты (основные характеристики), которые описывают особенности объекта
3. Модуль – это...
 - a. Средство для создания, изменения исходных файлов, которые содержат написанную на языке программирования программу
 - b. Автономно компилируемая программная единица, включающая в себя различные компоненты раздела описаний (типы, константы, переменные, процедуры и функции) и, возможно некоторые исполняемые операторы инициализирующей части**
 - c. Набор инструментов, которые используются для преобразования символов
4. Дополнить предложение. Набор всех файлов, необходимых для создания приложения, называется...
 - a. Модулем
 - b. Проектом**
 - c. Задачей
 - d. Формой
5. Файлы описания форм – это...
 - a. Отклики на внешнее воздействие
 - b. Файлы подключающие все используемые программные модули
 - c. Текстовые файлы с расширением DFM, в котором сохраняются значения свойств формы и ее компонентов**
6. Содержимое dfm-файла вызывается:
 - a. Контекстное меню формы -> View as Text**
 - b. Контекстное редактора кода -> View as Form
 - c. File -> Save as...
7. Дополнить предложение. При сборке приложения описание из DFM-файла помещается в...
 - a. Файл модуля
 - b. Область ресурсов**
 - c. Область установок проекта
8. Программный модуль (Unit) содержит:
 - a. Все относящиеся к форме объявления и методы обработки событий, написанные на языке Object Pascal**
 - b. Заготовку, макет одного из окон разрабатываемого приложения
 - c. Пиктограммы, которые представляют компоненты VCL, которые можно включить в приложение
9. Дополнить предложение. Имя модуля должно совпадать с...
 - a. Приложением Windows
 - b. Спецификацией задачи
 - c. Объектом в Delphi
 - d. Именем файла при сохранении File->Save as...**
10. Интерфейсная секция модуля (interface) содержит: (несколько вариантов ответа)
 - a. Список стандартных модулей библиотеки VCL**
 - b. Раздел описания типов**

- c. Компилятор
 - d. Раздел описания переменных (объявление собственно объекта формы)**
 - e. Инспектор объектов и проектировщик форм
11. Дополнить предложение. Класс формы объявляется в разделе...
- a. Type**
 - b. Uses
 - c. Project
 - d. Var
12. Дополнить предложение. В описании класса форм помещенные на форму компоненты представлены...
- a. Границами формы
 - b. Полями формы**
 - c. Индексами формы
13. Published – это...
- a. Атрибут видимости, позволяющий работать с компонентами и методами обработки событий на визуальном уровне**
 - b. Директива препроцессора
 - c. Текст программы на языке Object Pascal
14. Пустые секции, в которых можно размещать любые вспомогательные поля, методы и свойства: *(несколько вариантов ответа)*
- a. Private**
 - b. Uses
 - c. Var
 - d. Public**
 - e. Program
15. Объявление объекта формы:
- a. `unit Unit1;`
 - b. `type TForm1 = class(TForm)`
 - c. `var Form1: TForm1;`**

Тема 1 Отладка и тестирование программного обеспечения

Вопросы

1. Регрессионное тестирование: цели и задачи, условия применения, классификация тестов и методов отбора
2. Регрессионное тестирование: разновидности метода отбора тестов
3. Регрессионное тестирование: методики, не связанные с отбором тестов и методики порождения тестов
4. Регрессионное тестирование: алгоритм и программная система поддержки
5. Описание тестируемой системы и ее окружения. Планирование тестирования
6. Модульное тестирование на примере классов
7. Интеграционное тестирование
8. Системное тестирование
9. Ручное тестирование
10. Автоматизация тестирования с помощью скриптов
11. Автоматическая генерация тестов на основе формального описания
12. Описание ручного тестирования
13. Автоматизация тестирования с помощью скриптов

Тема 2 Документирование Практическая работа

Вариант №1

Даны две квадратные матрицы $n \times n$. Напечатать ту из них, которая имеет минимальный "след" (т.е. минимальную сумму элементов главной диагонали). При решении создать функцию для нахождения следа матрицы и функцию печати матрицы.

Вариант №2

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция px - номер максимального по модулю элемента массива; 2) если задана опция s - сумму элементов массива, расположенных после первого положительного элемента. Если заданы опции px и s - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №3

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция k - количество элементов массива, больших C (вводится с клавиатуры); 2) если задана опция pr - произведение элементов массива, расположенных после максимального по модулю элемента. Если заданы опции pr и k - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №4

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция ko - количество отрицательных элементов массива; 2) если задана опция s - сумму модулей элементов массива, расположенных после минимального по модулю элемента. Если заданы опции ko и s - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №5

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция kp - количество положительных элементов массива; 2) если задана опция $-s$ - сумму элементов массива, расположенных после последнего элемента, равного нулю. Если заданы опции kp и s - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №6

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция p - произведение отрицательных элементов массива; 2) если задана опция sm - сумму положительных элементов массива, расположенных до максимального элемента. Если заданы опции p и sm - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №7

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция k - количество элементов массива, равных 0; 2) если задана опция sp - сумму элементов массива, расположенных после минимального элемента. Если заданы опции k и sp - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №8

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция s - сумму элементов массива с нечетными номерами;

2) если задана опция *ss* - сумму элементов массива, расположенных между первым и последним отрицательными элементами. Если заданы опции *s* и *ss* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №9

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция *mx* - максимальный элемент массива; 2) если задана опция *s* - сумму элементов массива, расположенных до последнего положительного элемента. Если заданы опции *mx* и *s* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №10

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция *mn* - минимальный элемент массива; 2) если задана опция *s* - сумму элементов массива, расположенных между первым и последним положительными элементами. Если заданы опции *mn* и *s* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №11

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 целых элементов, вычислить: 1) если задана опция *n* - номер максимального элемента массива; 2) если задана опция *pr* - произведение элементов массива, расположенных между первым и вторым нулевыми элементами. Если заданы опции *pr* и *n* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №12

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция *nm* - номер минимального элемента массива; 2) если задана опция *s* - сумму элементов массива, расположенных между первым и вторым отрицательными элементами. Если заданы опции *nm* и *s* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №13

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция *mm* - максимальный по модулю элемент массива; 2) если задана опция *s* - сумму элементов массива, расположенных между первым и вторым положительными элементами. Если заданы опции *mm* и *s* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №14

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 целых элементов, вычислить: 1) если задана опция *mn* - минимальный по модулю элемент массива; 2) если задана опция *s* - сумму модулей элементов массива, расположенных после первого элемента, равного нулю. Если заданы опции *mn* и *s* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №15

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция *n* - номер минимального по модулю элемента массива; 2) если задана опция *sm* - сумму модулей элементов массива, расположенных после первого отрицательного элемента. Если заданы опции *n* и *sm* - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Вариант №16

В одномерном массиве, вводимом с клавиатуры и состоящем из 10 вещественных элементов, вычислить: 1) если задана опция `mm` - максимальный по модулю элемент массива; 2) если задана опция `s` - сумму элементов массива, расположенных между первым и вторым отрицательными элементами. Если заданы опции `mm` и `s` - выполнить оба задания; если опции не заданы – вывести сообщение и не выполнять задания.

Тема 2 Документирование

Тест

1- вариант

1. К комментариям не относятся: *(несколько вариантов ответа)*
 - 1) **Program**
 - 2) {Form1 }
 - 3) **{ \$R *.res }**
 - 4) //выполнение команды
2. Дополнить предложение. Сочетание `end` со следующей за ней точкой называется...
 - 1) Терминалом программной единицы
 - 2) **Терминатором программной единицы**
 - 3) Андромидом программной единицы
3. Содержимое DPR-файла вызывается: *(несколько вариантов ответа)*
 - 1) Ctrl+alt
 - 2) File -> New->Application
 - 3) **Project -> View Source**
 - 4) **Ctrl+F12**
4. Дополнить предложение. Зарезервированное слово **uses** сообщает компилятору о...
 - 1) **Модулях, которые должны рассматриваться как необъемлемые части программы и которые содержатся в других файлах**
 - 2) Начале работы приложения
 - 3) Ошибках, которые могут возникнуть при компилировании
5. Модуль Forms – это...
 - 1) Дополнительный модуль, содержащий неиспользуемые компоненты
 - 2) Модуль для построения более красочного приложения
 - 3) **Обязательный для всех программ модуль, который содержит определение объекта Application**
6. Запись `{ $R *.res }` означает:
 - 1) Комментарий, указывающий программисту где искать ресурсы
 - 2) **Директиву, подключающую к результирующему выполняемому файлу ресурсы**
 - 3) Директиву, подключающую к результирующему выполняемому файлу форму
7. Включение множественного выбора элементов компонента `ListBox1`
 - 1) `ListBox2.MultiSelect:=false;`
 - 2) **`ListBox1.MultiSelect:=true;`**
 - 3) `ListBox1.MultiSelect:=false;`
 - 4) `ListBox1.Checked:=true;`
8. Запись `ListBox1.Selected[3]:=true` означает что:
 - 1) Необходимо установить три дополнительных элемента
 - 2) Необходимо удалить три лишних элемента
 - 3) **Выделен четвертый элемент в компоненте**
 - 4) Выделен третий элемент в компоненте
9. Простейшая функция вывода на экран текстовых сообщений:
 - 1) `ListBox1.Items.Add();`

- 2) **ShowMessage();**
 - 3) Case...of;
 - 4) CheckBox1.Checked:=true;
10. Полная очистка списка:
- 1) **ListBox1.Clear;**
 - 2) ListBox1.Items.Add();
 - 3) RadioGroup1.ItemIndex:=0;
11. Снятие «флажка» компонента CheckBox1
- 1) ListBox1.Checked:=true;
 - 2) **CheckBox1.Checked:=false;**
 - 3) CheckBox1.Checked:=true;
 - 4) CheckBox1.Caption :='Снять флажок';
12. Свойство Align=AllBottom:
- 1) Компонент перемещается в центр формы
 - 2) Компонент перемещается в верхнюю часть контейнера
 - 3) **Компонент перемещается в нижнюю часть контейнера и его ширина становится равной ширине контейнера**
13. Объект ComboBox - ...
- 1) Многострочный редактор
 - 2) **Комбинированный список**
 - 3) Функциональная кнопка
14. Установка в ComboBox1 первой строки массива Item:
- 1) **ItemIndex=0**
 - 2) ItemIndex=-1
 - 3) ItemIndex=1
 - 4) ItemIndex=356
15. Объект RadioGroup - ...
- 1) Команда «Авторадио»
 - 2) **Панель группы радиокнопок**
 - 3) Метка

2-вариант

1. Дополнить предложение. Компонент-флажок, который используется для обозначения включения или отключения какой либо опции, называется...

- 1) RadioGroup
- 2) Функциональная кнопка
- 3) **CheckBox**
- 4) ListBox

2. Дополнить предложение. За стиль оформления внешней и внутренней рамок отвечают свойства ... (несколько вариантов ответа)

- 5) Align
- 6) **BevelOuter**
- 7) **BevelInner**
- 8) ssBoth
- 9) Caption

3. VCL – это...

- 10) Приложение Delphi
- 11) Алгоритмизация решения задачи
- 12) **Библиотека визуальных компонентов**
- 13) Характеристики объектов

4. Дополнить предложение. Компоненты, которые видны на форме, как во время создания приложения, так и во время работы приложения называются ...

- 14) **Визуальными**

- 15) Реальными
 - 16) Невизуальными
 - 17) Основными
5. К невидуальным компонентам относят:
- 18) Кнопки, метки, списки блоков
 - 19) Таймеры, компоненты для работы с базами данных, списки изображений**
 - 20) Библиотеки, звуки, коды
6. Дополнить предложение. Имя компонента состоит из... (несколько вариантов ответа)
- 21) Названия компонента**
 - 22) Заготовки компонента
 - 23) Порядкового номера компонента**
 - 24) Произвольного значения ItemIndex
7. implementation – это...
- 25) Тип данных Delphi
 - 26) Племя в Южной Африке
 - 27) Раздел реализации объявлений интерфейса**
8. Подключение файла описания формы:
- 28) {\$R *.dfm}**
 - 29) {\$R *.pas}
 - 30) {R *.dpr}
9. Дополнить предложение. После подключения файла описания формы следует...
- 31) Раздел описания типов
 - 32) Атрибут видимости
 - 33) Описание размеров используемых компонентов
 - 34) Реализация методов обработки событий**
10. Установить метод обработки события Button1Click: (несколько вариантов ответа)
- 35) Двойной щелчок по компоненту Button1 на форме**
 - 36) Выделить Button1 ->Events -> OnClick**
 - 37) Выделить BitBtn1 ->Events -> OnClick
11. Дополнить предложение. Инспектор объектов позволяет ... (несколько вариантов ответа)
- 1) Устанавливать свойства объектов**
 - 2) Изменять структуру программного кода
 - 3) Изменять наследование классов объектов
 - 4) Назначать методы обработки событий**
12. Дополнить предложение. Инспектор объектов состоит из ... (несколько вариантов ответа)
- 1) Unit – программного модуля
 - 2) Properties – списка свойств,**
 - 3) Events – списка событий.**
 - 4) Begin...end.
13. Окно содержащее текст программы на языке Object Pascal, связанный с каждой формой в приложении:
- 1) Окно компилятора
 - 2) Окно редактора кода**
 - 3) Окно проектировщика
 - 4) Главное окно
14. Задание метки текста из редактора ввода:
- 1) Memo1.Caption := Edit1.Text;
 - 2) Label1.Caption := Edit1.Text;**
 - 3) Form1.Caption := Edit1.Text;
15. Обнуление строки ввода:
- 1) Edit1.Text :='';**
 - 2) Edit1.Text := TMemo;
 - 3) Edit1.Text :='Закругляемся'

**Приложение 5
к рабочей программе**

Федеральное государственное бюджетное образовательное учреждение высшего образования
УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНЫ
на заседании Педагогического совета колледжа

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ

ПРОМЕЖУТОЧНОГО КОНТРОЛЯ

междисциплинарный курс

Поддержка и тестирование программных модулей

Билеты для зачета

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №1

Теоретические вопросы

1. Введение: тестирование - способ обеспечения качества программного продукта

2. Выберите один правильный ответ из предложенных вариантов.

а) В основные обязанности тестировщика входят:

- а. Выявление ошибки
- б. Исправление ошибки
- с. Объяснение причины ошибки

б) Одному тест-требованию может соответствовать:

- 1. только один тестовый пример
- 2. несколько тестовых примеров
- 3. не более двух тестовых примеров

с) Два тестовых примера проверяют один и тот же класс эквивалентности:

- а. если от них получен один и тот же результат
- б. если от них получена одинаковая реакция системы
- с. если они построены по одному тест-требованию

Практическое задание

Разработать программу, которая принимает от пользователя квадратную матрицу, вычисляет сумму элементов, лежащих выше главной диагонали. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет № 2

Теоретические вопросы

1. Основные понятия тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований
2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая принимает от пользователя квадратную матрицу, вычисляет сумму элементов, лежащих выше побочной диагонали. Разработать пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №3

Теоретические вопросы

1. Подходы к обоснованию истинности формул и программ и их связь с тестированием. Вопросы организации тестирования. Фазы тестирования, основные проблемы тестирования и поставлена задача выбора конечного набора тестов.

2. Выберите один правильный ответ из предложенных вариантов.

а) Согласно методу МС\ДС для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. 4-х тестовых примеров
3. 5-х тестовых примеров

б) Одной из основных задач анализа полноты покрытия кода является:

1. выявление участков кода, которые выполняются при выполнении тестовых примеров
2. выявление участков кода, которые содержат ошибки
3. выявление участков кода, которые не выполняются при выполнении тестовых примеров

с) При использовании какого метода интеграционного тестирования сначала все программные модули, входящие в состав системы, тестируются и только затем объединяются для интеграционного тестирования?

1. восходящего
2. монолитного
3. нисходящего

Практическое задание

Разработать программу, позволяющую переводить двоичные целые числа в десятичные и, наоборот, десятичные – в двоичные. Предусмотреть пользовательский интерфейс.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №4

Теоретические вопросы

1. Требования к идеальному критерию тестирования и классы частных критериев. Особенности применения структурных и функциональных критериев на базе конкретных примеров. Особенности применения методов стохастического тестирования и метод оценки скорости выявления ошибок. Мутационный критерий и на примере иллюстрируется техника работы с ним.

2. Выберите один правильный ответ из предложенных вариантов.

а) При использовании какого метода интеграционного тестирования подразумевается, что, как только разрабатывается новый модуль системы, он сразу же интегрируется со всей остальной системой?

1. восходящего
2. монолитного
3. с постоянной интеграцией

б) Описательные спецификации могут быть ...

1. последовательными
2. параллельными
3. алгебраическими

с) При тестировании методом черного ящика используются следующие критерии:

1. графа причин и следствий
2. покрытия операторов
3. покрытия ребер

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет сумму отрицательных элементов массива, произведение элементов массива, расположенных между максимальным и минимальным элементами. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №5

Теоретические вопросы

1. Оценка оттестированности проекта: метрики и методика интегральной оценки

2. Выберите один правильный ответ из предложенных вариантов.

а) Предусмотрение изменений – это принцип, который влияет на такие качества программного обеспечения как ...

1. повторную применимость
2. прозрачность
3. детерминированность реализации

б) Методы и технологии реинжиниринга и обратного инжиниринга программного обеспечения нацелены на ...

1. тестирование унаследованного программного обеспечения
2. оптимизацию унаследованного программного обеспечения
3. модификацию унаследованного программного обеспечения

в) С точки зрения разработчика программного обеспечения качество последнего заключается в ...

1. производительности
2. легкости применения
3. переносимости

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет сумму положительных элементов массива, произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №6

Теоретические вопросы

1. Графовые модели проекта, метрики оценки оттестированности проекта, приводятся примеры плоской и иерархической моделей проекта

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований
2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n целых элементов, преобразовывает массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №7

Теоретические вопросы

1. Особенности модульного тестирования, подходы к тестированию на основе потока управления, потока данных, динамические и статические методы при структурном подходе. Взаимосвязь сборки модулей и методов интеграционного тестирования. Подходы монолитного, инкрементального, нисходящего и восходящего тестирования. Рассматриваются особенности интеграционного тестирования в процедурном программировании.

2. Выберите один правильный ответ из предложенных вариантов.

а) В основные обязанности тестировщика входят:

1. Выявление ошибки
2. Исправление ошибки
3. Объяснение причины ошибки

б) Одному тест-требованию может соответствовать:

1. только один тестовый пример
2. несколько тестовых примеров
3. не более двух тестовых примеров

с) Два тестовых примера проверяют один и тот же класс эквивалентности:

- a. если от них получен один и тот же результат
- b. если от них получена одинаковая реакция системы
- c. если они построены по одному тест-требованию

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n целых элементов, преобразовывает массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №8

Теоретические вопросы

1. Интеграционное тестирование и его особенности для объектно-ориентированного программирования

2. Выберите один правильный ответ из предложенных вариантов.

а) При использовании какого метода интеграционного тестирования подразумевается, что, как только разрабатывается новый модуль системы, он сразу же интегрируется со всей остальной системой?

1. восходящего
2. монолитного
3. с постоянной интеграцией

б) Описательные спецификации могут быть ...

1. последовательными
2. параллельными
3. алгебраическими

с) При тестировании методом черного ящика используются следующие критерии:

1. графа причин и следствий
2. покрытия операторов
3. покрытия ребер

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет количество элементов массива, больших C , произведение элементов массива, расположенных после максимальным по модулю элемента. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №9

Теоретические вопросы

1. Модель объектно-ориентированной программы, использующая понятие Р-путей и ММ-путей. Оценки сложности тестирования и методика тестирования объектно-ориентированной программы. Рассматривается пример интеграционного тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Согласно методу МС\ДС для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. 4-х тестовых примеров
3. 5-х тестовых примеров

б) Одной из основных задач анализа полноты покрытия кода является:

1. выявление участков кода, которые выполняются при выполнении тестовых примеров
2. выявление участков кода, которые содержат ошибки
3. выявление участков кода, которые не выполняются при выполнении тестовых примеров

с) При использовании какого метода интеграционного тестирования сначала все программные модули, входящие в состав системы, тестируются и только затем объединяются для интеграционного тестирования?

1. восходящего
2. монолитного
3. нисходящего

3. Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, упорядочивает по возрастанию отдельно элементы, стоящие на четных местах, и элементы, стоящие на нечетных местах. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №10

Теоретические вопросы

1. Разновидности тестирования: системное и регрессионное тестирование

2. Выберите один правильный ответ из предложенных вариантов.

а) Предусмотрение изменений – это принцип, который влияет на такие качества программного обеспечения как ...

1. повторную применимость
2. прозрачность
3. детерминированность реализации

б) Методы и технологии реинжиниринга и обратного инжиниринга программного обеспечения нацелены на ...

1. тестирование унаследованного программного обеспечения
2. оптимизацию унаследованного программного обеспечения
3. модификацию унаследованного программного обеспечения

в) С точки зрения разработчика программного обеспечения качество последнего заключается в ...

1. производительности
2. легкости применения
3. переносимости

Практическое задание

Разработать программу, которая принимает от пользователя целочисленную квадратную матрицу, определяет сумму элементов в тех строках, которые не содержат отрицательных элементов. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №11

Теоретические вопросы

1. Автоматизация тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований

2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая определяет, сколько раз встретилось заданное слово в текстовом файле, длина строки в котором не превышает 80 символов. Текст не содержит переносов слов. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №12

Теоретические вопросы

1. Особенности индустриального тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Предусмотрение изменений – это принцип, который влияет на такие качества программного обеспечения как ...

1. повторную применимость
2. прозрачность
3. детерминированность реализации

б) Методы и технологии реинжиниринга и обратного инжиниринга программного обеспечения нацелены на ...

1. тестирование унаследованного программного обеспечения
2. оптимизацию унаследованного программного обеспечения
3. модификацию унаследованного программного обеспечения

в) С точки зрения разработчика программного обеспечения качество последнего заключается в ...

1. производительности
2. легкости применения
3. переносимости

Практическое задание

Разработать программу, которая считывает текст из файла и выводит на экран только вопросительные предложения из этого текста. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №13

Теоретические вопросы

1. Документирование и оценка индустриального тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Согласно методу MC\DC для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. 4-х тестовых примеров
3. 5-х тестовых примеров

б) Одной из основных задач анализа полноты покрытия кода является:

1. выявление участков кода, которые выполняются при выполнении тестовых примеров
2. выявление участков кода, которые содержат ошибки
3. выявление участков кода, которые не выполняются при выполнении тестовых примеров

с) При использовании какого метода интеграционного тестирования сначала все программные модули, входящие в состав системы, тестируются и только затем объединяются для интеграционного тестирования?

1. восходящего
2. монолитного
3. нисходящего

Практическое задание

Разработать программу, которая считывает текст из файла и выводит на экран только предложения, не содержащие запятых. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №14

Теоретические вопросы

1. Описываются особенности документирования тестовых процедур для ручных и автоматизированных тестов, описаний тестовых наборов и тестовых отчетов. Рассматривается жизненный цикл дефекта. Обсуждаются метрики, используемые при тестировании

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований

2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая считывает текст из файла и выводит на экран предложения, содержащие максимальное количество знаков пунктуации. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №15

Теоретические вопросы

1. Регрессионное тестирование: цели и задачи, условия применения, классификация тестов и методов отбора

2. Выберите один правильный ответ из предложенных вариантов.

а) В основные обязанности тестировщика входят:

1. Выявление ошибки
2. Исправление ошибки
3. Объяснение причины ошибки

б) Одному тест-требованию может соответствовать:

1. только один тестовый пример
2. несколько тестовых примеров
3. не более двух тестовых примеров

3. Два тестовых примера проверяют один и тот же класс эквивалентности:

1. если от них получен один и тот же результат
2. если от них получена одинаковая реакция системы
3. если они построены по одному тест-требованию

Практическое задание

Разработать программу, которая считывает текст из файла и выводит на экран сначала вопросительные, а затем восклицательные предложения. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №16

Теоретические вопросы

1. Введение: тестирование - способ обеспечения качества программного продукта

2. Выберите один правильный ответ из предложенных вариантов.

а) Согласно методу MC\DC для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. 4-х тестовых примеров
3. 5-х тестовых примеров

б) Одной из основных задач анализа полноты покрытия кода является:

1. выявление участков кода, которые выполняются при выполнении тестовых примеров
2. выявление участков кода, которые содержат ошибки
3. выявление участков кода, которые не выполняются при выполнении тестовых примеров

с) При использовании какого метода интеграционного тестирования сначала все программные модули, входящие в состав системы, тестируются и только затем объединяются для интеграционного тестирования?

1. восходящего
2. монолитного
3. нисходящего

Практическое задание

Разработать программу, которая принимает от пользователя квадратную матрицу, вычисляет сумму элементов, лежащих выше главной диагонали. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет № 17

Теоретические вопросы

1. Основные понятия тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований
2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая принимает от пользователя квадратную матрицу, вычисляет сумму элементов, лежащих выше побочной диагонали. Разработать пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №18

Теоретические вопросы

1. Подходы к обоснованию истинности формул и программ и их связь с тестированием. Вопросы организации тестирования. Фазы тестирования, основные проблемы тестирования и поставлена задача выбора конечного набора тестов.

2. Выберите один правильный ответ из предложенных вариантов.

а) При использовании какого метода интеграционного тестирования подразумевается, что, как только разрабатывается новый модуль системы, он сразу же интегрируется со всей остальной системой?

1. восходящего
2. монолитного
3. с постоянной интеграцией

б) Описательные спецификации могут быть ...

1. последовательными
2. параллельными
3. алгебраическими

с) При тестировании методом черного ящика используются следующие критерии:

1. графа причин и следствий
2. покрытия операторов
3. покрытия ребер

Практическое задание

Разработать программу, позволяющую переводить двоичные целые числа в десятичные и, наоборот, десятичные – в двоичные. Предусмотреть пользовательский интерфейс.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №19

Теоретические вопросы

1. Требования к идеальному критерию тестирования и классы частных критериев. Особенности применения структурных и функциональных критериев на базе конкретных примеров. Особенности применения методов стохастического тестирования и метод оценки скорости выявления ошибок. Мутационный критерий и на примере иллюстрируется техника работы с ним.

2. Выберите один правильный ответ из предложенных вариантов.

а) Тестовое окружение для программного кода на структурных языках программирования состоит из:

1. драйвера
2. тестов
3. исходного кода

б) Полная система тестов позволяет утверждать, что:

1. система реализует всю функциональность, указанную в требованиях
2. система работает корректно
3. система работает правильно

в) Согласно методу MC\DC для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. 4-х тестовых примеров
3. 5-х тестовых примеров

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет сумму отрицательных элементов массива, произведение элементов массива, расположенных между максимальным и минимальным элементами. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №20

Теоретические вопросы

1. Оценка оттестированности проекта: метрики и методика интегральной оценки

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований
2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет сумму положительных элементов массива, произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №21

Теоретические вопросы

1. Графовые модели проекта, метрики оценки оттестированности проекта, приводятся примеры плоской и иерархической моделей проекта

2. Выберите один правильный ответ из предложенных вариантов.

а) В основные обязанности тестировщика входят:

1. Выявление ошибки
2. Исправление ошибки
3. Объяснение причины ошибки

б) Одному тест-требованию может соответствовать:

1. только один тестовый пример
2. несколько тестовых примеров
3. не более двух тестовых примеров

с) Два тестовых примера проверяют один и тот же класс эквивалентности:

- a. если от них получен один и тот же результат
- b. если от них получена одинаковая реакция системы
- c. если они построены по одному тест-требованию

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n целых элементов, преобразовывает массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №22

Теоретические вопросы

1. Особенности модульного тестирования, подходы к тестированию на основе потока управления, потока данных, динамические и статические методы при структурном подходе. Взаимосвязь сборки модулей и методов интеграционного тестирования. Подходы монолитного, инкрементального, нисходящего и восходящего тестирования. Рассматриваются особенности интеграционного тестирования в процедурном программировании.

2. Выберите один правильный ответ из предложенных вариантов.

а) При использовании какого метода интеграционного тестирования подразумевается, что, как только разрабатывается новый модуль системы, он сразу же интегрируется со всей остальной системой?

1. восходящего
2. монолитного
3. с постоянной интеграцией

б) Описательные спецификации могут быть ...

1. последовательными
2. параллельными
3. алгебраическими

с) При тестировании методом черного ящика используются следующие критерии:

1. графа причин и следствий
2. покрытия операторов
3. покрытия ребер

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n целых элементов, преобразовывает массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №23

Теоретические вопросы

1. Интеграционное тестирование и его особенности для объектно-ориентированного программирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Тестовое окружение для программного кода на структурных языках программирования состоит из:

1. драйвера
2. тестов
3. исходного кода

б) Полная система тестов позволяет утверждать, что:

1. система реализует всю функциональность, указанную в требованиях
2. система работает корректно
3. система работает правильно

с) Согласно методу MC\DC для тестирования логической функции с тремя входами и одним выходом достаточно:

1. 3-х тестовых примеров
2. 4-х тестовых примеров
3. 5-х тестовых примеров

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, вычисляет количество элементов массива, больших C , произведение элементов массива, расположенных после максимальным по модулю элемента. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №24

Теоретические вопросы

1. Модель объектно-ориентированной программы, использующая понятие Р-путей и ММ-путей. Оценки сложности тестирования и методика тестирования объектно-ориентированной программы. Рассматривается пример интеграционного тестирования

2. Выберите один правильный ответ из предложенных вариантов.

а) Модульное тестирование проводится для того, чтобы:

1. удостовериться в корректной работе системы в целом
2. удостовериться в корректной работе набора модулей
3. удостовериться в корректной работе отдельного модуля

б) Модуль – это (с точки зрения наших семинарских занятий):

1. часть программного кода, выполняющая одну функцию с точки зрения функциональных требований

2. программный модуль, т.е. минимальный компилируемый элемент программной системы
3. задача в списке задач проекта

с) Какие основные задачи решаются в ходе модульного тестирования?

1. Поиск и документирование несоответствий требованиям
2. Рефакторинг модулей
3. Отладка

Практическое задание

Разработать программу, которая принимает от пользователя массив, состоящий из n вещественных элементов, упорядочивает по возрастанию отдельно элементы, стоящие на четных местах, и элементы, стоящие на нечетных местах. Предусмотреть пользовательский интерфейс для программы.

УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

| | |
|------------------------------|--|
| Междисциплинарный курс | Поддержка и тестирование программных модулей |
| Вид промежуточной аттестации | Зачет |
| Составил | Кольева Н.А. |

Билет №25

Теоретические вопросы

1. Разновидности тестирования: системное и регрессионное тестирование

2. Выберите один правильный ответ из предложенных вариантов.

а) В основные обязанности тестировщика входят:

- а. Выявление ошибки
- б. Исправление ошибки
- с. Объяснение причины ошибки

б) Одному тест-требованию может соответствовать:

- 1. только один тестовый пример
- 2. несколько тестовых примеров
- 3. не более двух тестовых примеров

с) Два тестовых примера проверяют один и тот же класс эквивалентности:

- а. если от них получен один и тот же результат
- б. если от них получена одинаковая реакция системы
- с. если они построены по одному тест-требованию

Практическое задание

Разработать программу, которая принимает от пользователя целочисленную квадратную матрицу, определяет сумму элементов в тех строках, которые не содержат отрицательных элементов. Предусмотреть пользовательский интерфейс для программы.