

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лкодинович
Должность: Ректор
Дата подписания: 03.06.2022 09:54:55
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Дагестанский государственный технический университет»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

МДК.01.02 Поддержка и тестирование программных модулей

специальность

09.02.07 Информационные системы и
программирование

квалификация

программист

уровень образования

СПО на базе основного общего образования/
среднего общего образования

Разработчик



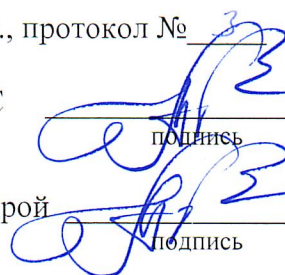
подпись

Айгумов Т.Г., к.э.н., доцент

Фонд оценочных средств обсужден на заседании кафедры ПОВТиАС

« 1 » 11 2022 г., протокол № 3

Зав. кафедрой ПОВТиАС



подпись

Айгумов Т.Г., к.э.н., доцент

Зав. выпускающей кафедрой



подпись

Айгумов Т.Г., к.э.н., доцент

г. Махачкала - 2022

СОДЕРЖАНИЕ

| | |
|---|----|
| 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ | 3 |
| 2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ | 3 |
| 3. ОЦЕНКА ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ..... | 5 |
| 3.1. Контроль и оценка освоения учебной дисциплины по темам | 5 |
| 3.2. Перечень заданий для текущего контроля..... | 7 |
| 4. ПЕРЕЧЕНЬ ЗАДАНИЙ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ.. | 12 |
| 5. КРИТЕРИИ ОЦЕНКИ..... | 21 |

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы дисциплины МДК.01.02 «Поддержка и тестирование программных модулей» и предназначен для контроля и оценки образовательных достижений, обучающихся (в т.ч. по самостоятельной работе студентов, далее – СРС), освоивших программу данной дисциплины.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям ФГОС СПО по специальности 09.02.07 Информационные системы и программирование.

Рабочей программой дисциплины МДК.01.02 «Поддержка и тестирование программных модулей» предусмотрено формирование профессиональных компетенций:

- 1) ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;
 - 2) ПК 1.4 Выполнять тестирование программных модулей;
 - 3) ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода.
- Формой аттестации по учебной дисциплине является зачет.

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

В результате аттестации по учебной дисциплине осуществляется комплексная проверка следующих умений и знаний, а также динамика формирования профессиональных компетенций.

Таблица 1

| Результаты обучения: знания, умения, практический опыт | Формируемые виды деятельности/ компетенции |
|--|--|
| Знать: З1 - актуальная нормативно-правовая база в области документирования алгоритмов | Разработка модулей программного обеспечения для компьютерных систем/ПК 1.1 |
| Уметь: У1 - оценка сложности алгоритма | |
| Иметь практический опыт в: П1- разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования. | |
| Знать: З1 - основные виды и принципы тестирования программных продуктов | |
| Уметь: У1 - выполнять отладку и тестирование программы на уровне модуля; У2 - оформлять документацию на программные средства | Разработка модулей программного обеспечения для компьютерных систем/ПК 1.4 |
| Иметь практический опыт в: П1 - проводить тестирование программного модуля по определенному сценарию; | |
| П2 - использовать инструментальные средства на этапе тестирования программного продукта. | |
| Знать: | |

| Результаты обучения: знания, умения, практический опыт | Формируемые виды деятельности/ компетенции |
|--|--|
| 31 - способы оптимизации и приемы рефакторинга; | Разработка модулей программного обеспечения для компьютерных систем/ПК 1.5 |
| 32 - методы организации рефакторинга и оптимизации кода; | |
| 33 - принципы работы с системой контроля версий. | |
| Уметь: | |
| У1 - выполнять оптимизацию и рефакторинг программного кода; | |
| У2 - работать с системой контроля версий. | |
| Иметь практический опыт: | |
| П1 - осуществлять рефакторинг и оптимизацию программного кода. | |

3. ОЦЕНКА ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Контроль и оценка освоения учебной дисциплины по темам

Предметом оценки служат знания, умения и практический опыт, предусмотренные ФГОС СПО, направленные на формирование профессиональных компетенций.

Таблица 2

Контроль и оценка освоения учебной дисциплины по темам

| Элемент учебной дисциплины | Формы и методы контроля | | | |
|--|---|---|--------------------------|---|
| | Текущий контроль | | Промежуточная аттестация | |
| | Форма контроля | Проверяемые компетенции/знания/умения/ практический опыт | Форма контроля | Проверяемые компетенции/знания/умения / практический опыт |
| Тема 1. Отладка программных модулей. | Письменная работа; Устный опрос; Практическая работа №1; Практическая работа №2; Практическая работа №3; Лабораторная работа №1; Лабораторная работа №2; Лабораторная работа №3; Самостоятельная работа | ПК 1.1, 31, У1, П1; ПК 1.4, 31, У1, П1 | Зачетная работа | ПК 1.1, 31, У1, П1; ПК 1.4, 31, У1, П1 |
| Тема 2. Отладка и тестирование программного продукта на уровне модулей. | Письменная работа; Устный опрос; Практическая работа №4; Практическая работа №5; Лабораторная работа №4; Лабораторная работа №5; Лабораторная работа №6; | ПК 1.4, 31, У1, П1, П2; ПК 1.5, 31, 32, 33, У1, У2, П1 | Зачетная работа | ПК 1.4, 31, У1, П1, П2; ПК 1.5, 31, 32, 33, У1, У2, П1 |

| Элемент учебной дисциплины | Формы и методы контроля | | | |
|---------------------------------|---|--|--------------------------|---|
| | Текущий контроль | | Промежуточная аттестация | |
| | Форма контроля | Проверяемые компетенции/знания/умения/ практический опыт | Форма контроля | Проверяемые компетенции/знания/умения / практический опыт |
| | Лабораторная работа №7; Самостоятельная работа | | | |
| Тема 3. Документирование ПО. | Письменная работа; Устный опрос; Практическая работа №6; Практическая работа №7; Лабораторная работа №8; Лабораторная работа №9; Лабораторная работа №10; Самостоятельная работа | ПК 1.1, 31, У1, П1; ПК 1.4, 31, У2, П1, П2 | Зачетная работа | ПК 1.1, 31, У1, П1; ПК 1.4, 31, У2, П1, П2 |

3.2. Перечень заданий для текущего контроля

Формируемая компетенция: ПК 1.1

Перечень заданий закрытого типа

Задание №1. Какой стандарт распространяется на условные обозначения (символы) в схемах алгоритмов, программ, данных и систем и устанавливает правила выполнения схем, используемых для отображения различных видов задач обработки данных и средств их решения?

- а) ISO 12207 - Информационные технологии. Процессы жизненного цикла программного обеспечения;
- б) ISO 6592 – Обработка информации. Руководство по разработке документации для вычислительных систем;
- в) ГОСТ 19.701-90 «ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила»;
- г) ГОСТ 34.602—89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.

Задание №2. Какое значение сохранится в переменной length, если на вход программы подать число 1024?

```
n = int(input())
length = 0
while n > 0:
    n //= 10
    length += 1
```

- а) 2;
- б) 5;
- в) 10;
- г) 4.

Задание №3. Сопоставьте виды сложности алгоритмов и их краткие определения.

| | |
|--|--|
| 1. Константная ($O(1)$); | а) удвоение размера входных данных увеличивает время выполнения в 4 раза. |
| 2. Линейная ($O(n)$); | б) сложность алгоритма растёт логарифмически с увеличением входных данных. |
| 3. Логарифмическая ($O(\log n)$); | в) сложность алгоритма линейно растёт с увеличением входных данных. |
| 4. Квадратичная ($O(n^2)$, $O(n^2)$). | г) вычислительная сложность алгоритма не зависит от входных данных. |

Задание №4. Сопоставьте виды программных документов и их содержание.

| | |
|---|------------------------------------|
| 1. Сведения о логической структуре и функционировании программы | а) техническое задание; |
| 2. Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля | б) пояснительная записка; |
| 3. Назначение и область применения программы, технические, технико-экономические и специальные требования, предъявляемые к программе, необходимые стадии и сроки разработки, виды испытаний | в) описание программы; |
| 4. Схема алгоритма, общее описание алгоритма и (или) функционирования программы, а также обоснование принятых технических и технико-экономических решений | г) программа и методика испытаний. |

Задание №5. Установите правильную последовательность этапов проектирования и разработки программ.

- а) разработка алгоритма;
- б) документирование;
- в) постановка задачи;
- г) построение модели;
- д) тестирование программы;
- е) написание программы;
- ж) проектирование программы;
- з) отладка программы.

Перечень заданий открытого типа

Задание №1. Какой блок используется для обозначения переходов управления по условию?

Задание №2. Как называется абстрактный исполнитель (абстрактная вычислительная машина), предложенная для формализации понятия алгоритма?

Задание №3. Как называется описание структуры алгоритма на специальном искусственном языке?

Задание №4. Дополните предложение: «Алгоритмы конечны, они должны завершаться и выдавать ... за заранее известное число шагов».

Задание №5. Дополните предложение: «Сложность алгоритма представляется как функция $f(n)$, где n — размер ... данных».

Формируемая компетенция: ПК 1.4

Перечень заданий закрытого типа

Задание №1. Как называется набор входных значений, предусловий выполнения, ожидаемых результатов и постусловий выполнения, разработанный для определенной цели или тестового условия, таких как выполнение определенного пути программы, либо для проверки соответствия определенному требованию?

- а) тестовый сценарий;

- б) тестовый набор;
- в) тестовый массив;

Задание №2. В какой блок помещается основной код, в котором потенциально может возникнуть исключение в конструкции try..except в Python?

- а) после ключевого слова finally;
- б) после ключевого слова try;
- в) после ключевого слова except;
- г) после ключевого слова else.

Задание №3. Сопоставьте между собой типы ошибок, выявленные при тестировании ПО, и характеристики ошибок.

| | |
|---|-------------------------|
| 1. Ошибки, обнаруженные компоновщиком при объединении модулей программы, называют | а) ошибками выполнения; |
| 2. Ошибки, фиксируемые компилятором при выполнении синтаксического и частично семантического анализа программы называют | б) ошибками компоновки; |
| 3. Ошибки кодирования называют | в) ошибками компиляции. |

Задание №4. Сопоставьте название уровня тестирования программного обеспечения и их краткие характеристики.

| | |
|-------------------------|---|
| 1. Модульный тест; | а) предназначен для проверки связи между модулями, помогает выявить ошибки, которые мешают взаимодействовать объединенным компонентам; |
| 2. Системный тест; | б) позволяет проверить компоненты на уровне модулей, тестировщик проверяет каждый исходный код и сопоставляет его с ожидаемым результатом; |
| 3. Интеграционный тест. | в) с помощью него проверяют всё программное обеспечение, даёт полный отчёт о производительности и соответствии системы заданным бизнес-требованиям; |

Задание №5. Установите правильную последовательность содержания рекомендованного тестового сценария?

- а) Постусловие (Post Conditions);
- б) Описание (Summary);
- в) Предусловие (Pre Conditions);
- г) Ожидаемый результат (Expected Result);
- д) Шаги (степы) (Steps);
- е) Уникальное краткое название (ID);
- ж) Статус (Pass/Fail);

Перечень заданий открытого типа

Задание №1. Как называется процесс проверки соответствия программного обеспечения требованиям, осуществляемый с помощью наблюдения за его работой в специальных, искусственно построенных ситуациях?

Задание №2. Как называется методика, изучающая функциональность ПО без необходимости знания внутренней структуры кода?

Задание №3. Как называется точное, однозначное, недвусмысленное задание для программиста, написанное постановщиком задачи?

Задание №4. Укажите номер стандарта: «ГОСТ «Виды программ и программных документов» устанавливает виды программ и программных документов для вычислительных машин, комплексов и систем независимо от их назначения и области применения».

Задание №5. Дополните определение: «Определение соответствия разрабатываемого ПО ожиданиям и потребностям пользователя, требованиям к системе называется ... ».

Формируемая компетенция: ПК 1.5

Перечень заданий закрытого типа

Задание №1. Что такое оптимизация программ?

- а) создание удобного интерфейса пользователя;
- б) разработка модульной конструкции программы;
- в) модификация программы с целью повышения эффективности ее работы;
- г) применение методов объектно-ориентированного программирования.

Задание №2. Что такое Git?

- а) язык программирования;
- б) система управления версиями;
- в) система управления базами данных;
- г) графический интерфейс для программирования.

Задание №3. Сопоставьте названия проблем программного кода и их характеристики.

| | |
|--|--|
| 1. Переменная, параметр, метод или класс больше не используется: требования к программе изменились. | а) некорректность имен переменных, функций, классов; |
| 2. Один и тот же программный код выполняет одно и то же действие в нескольких местах программы. | б) мёртвый код; |
| 3. Имена должны сообщать, почему элемент программного кода существует, что он делает и как используется. | в) дублирование. |

Задание №4. Сопоставьте названия методов оптимизации программного кода и их формулировки.

| | |
|-----------------|---|
| 1. Меморизация; | а) это метод замены строгого алгоритма на наиболее подходящие приближенные значения, что влечет за собой определенную потерю точности вычислений; |
| 2. Кеширование; | б) это простое сохранение результата выполнения определенной функции, которое поможет избежать ее повторного выполнения; |
| 3. Приближение. | в) это метод временного хранения данных в памяти устройства пользователя; |

Задание №5. Установите правильную последовательность этапов создания версии документа в системе контроля версий.

- а) ветвление;

- б) извлечение;
- в) обновление рабочей копии;
- г) слияние;
- д) фиксация изменений.

Перечень заданий открытого типа

Задание №1. Как называется процесс изменения внутренней структуры программного продукта, не затрагивающий её внешнего поведения и имеющий целью облегчение понимания программного кода?

Задание № 2. Как можно упростить предложенный фрагмент программного кода на Python, создающий список из строки, применив конструктор списков?

```
list = []
for char in 'abcdef':
    if char != 'c':
        list.append(char * 2)
print(list)
```

Задание №3. В предложенном фрагменте программного кода выполните рефакторинг строки вывода результата.

```
def input_value (prompt):
    while True:
        value = int( input( prompt ) )
        if value < 0 :
            print ("Значение не должно быть отрицательным!")
        else :
            return value
```

```
price = input_value ("Введите цену товара: ")
quantity = input_value ("Введите количество товара: ")
cost = price * quantity
print("Стоимость товара: ", cost)
```

Задание №4. Дополните определение: «В теории компиляторов, ... кодом называют код, который может быть исполнен, исполнялся или мог быть исполнен ранее, но результаты его вычислений не влияют на дальнейшую программу».

Задание №5. Дополните определение словосочетанием: «... версий – это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определенной версии».

4. ПЕРЕЧЕНЬ ЗАДАНИЙ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Формируемая компетенция: ПК 1.1

Задание №1. Какой стандарт распространяется на условные обозначения (символы) в схемах алгоритмов, программ, данных и систем и устанавливает правила выполнения схем, используемых для отображения различных видов задач обработки данных и средств их решения?

- а) ISO 12207 - Информационные технологии. Процессы жизненного цикла программного обеспечения;
- б) ISO 6592 – Обработка информации. Руководство по разработке документации для вычислительных систем;
- в) ГОСТ 19.701-90 «ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила»;
- г) ГОСТ 34.602—89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.

Задание №2. Какой блок используется для обозначения переходов управления по условию?

- а) процесс;
- б) решение;
- в) пуск-останов;
- г) предопределенный процесс;
- д) документ.

Задание №3. Как называется описание структуры алгоритма на специальном искусственном языке?

- а) блок-схема;
- б) граф;
- в) псевдокод
- г) программа.

Задача №4. Какой стандарт распространяется на системы для автоматизации различных видов деятельности (управление, проектирование, исследование и т. п.), включая их сочетания, и устанавливает состав, содержание, правила оформления документа “Техническое задание на создание (развитие или модернизацию) системы”?

- а) ГОСТ 34.602—89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы;
- б) ГОСТ Р 51904-2002 — Программное обеспечение встроенных систем. Общие требования к разработке и документированию;
- в) ISO 12207 - Информационные технологии. Процессы жизненного цикла программного обеспечения;
- г) ISO 6592 – Обработка информации. Руководство по разработке документации для вычислительных систем.

Задание №5. Как называется таблица, в которой каждому данному присвоено уникальное имя, задан тип, начальное значение данного и краткий комментарий его назначения в предметной области программирования?

- а) библиотека модулей;

- б) приложение данных;
- в) словарь данных;
- г) каталог данных.

Задание №6. Какой критерий качества алгоритмов определяется количеством промежуточных результатов, которые должны одновременно храниться в памяти исполнителя?

- а) разветвленность алгоритма
- б) объем алгоритма;
- в) длительность решения;
- г) связность алгоритма;
- д) цикличность алгоритма.

Задание №7. Сопоставьте характеристики и названия схем, используемых для отображения различных видов задач обработки данных и средств их решения.

| | |
|--|-----------------------------------|
| 1. отображают путь данных при решении задач и определяют этапы обработки, а также различные применяемые носители данных. | а) схемы данных; |
| 2. отображают последовательность операций в программе. | б) схемы работы системы; |
| 3. отображают управление операциями и поток данных в системе. | в) схемы программ; |
| 4. отображают путь активации программ и взаимодействий с соответствующими данными | г) схемы взаимодействия программ. |

Задание №8. Сопоставьте виды программных документов и их содержание.

| | |
|---|------------------------------------|
| 1. Сведения о логической структуре и функционировании программы | а) техническое задание; |
| 2. Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля | б) пояснительная записка; |
| 3. Назначение и область применения программы, технические, технико-экономические и специальные требования, предъявляемые к программе, необходимые стадии и сроки разработки, виды испытаний | в) описание программы; |
| 4. Схема алгоритма, общее описание алгоритма и (или) функционирования программы, а также обоснование принятых технических и технико-экономических решений | г) программа и методика испытаний. |

Задание №9. Установите правильную последовательность этапов проектирования и разработки программ.

- а) разработка алгоритма;
- б) документирование;
- в) постановка задачи;
- г) построение модели;
- д) тестирование программы;
- е) написание программы;
- ж) проектирование программы;
- з) отладка программы.

Задание №10. Установите правильную последовательность строк предложенного программного решения задачи: «Дан одномерный массив A числовых

значений, насчитывающий N элементов. Найти среднее арифметическое элементов массива».

- а) `print ('Результат: ', SR);`
- б) `SR=S/N;`
- в) `ch= int (input('Введите число: '));`
- г) `for i in range (N-1)::`
- д) `S=S+ch;`
- е) `N =int (input('Количество элементов: '));`
- ж) `S=0.`

Перечень заданий открытого типа

Задание №1. Как называется среда программирования, состоящая из общего набора данных (словарь данных) и набора программных модулей (словарь и библиотека программных модулей).

Задание №2. Как называют в теории алгоритмов множество вычислительных задач с примерно одинаковыми по сложности вычислениями?

Задание №3. Как называется абстрактный исполнитель (абстрактная вычислительная машина), предложенная для формализации понятия алгоритма?

Задание №4. Как называется алгоритм, если его сложность $t(n)$ в наихудшем случае ограничена сверху некоторым многочленом от n ?

Задание №5. Какой критерий качества алгоритмов определяется количеством шагов алгоритма, а также сложностью этих шагов?

Задание №6. Как называется величина, способная изменяться в процессе выполнения программы?

Задание №7. Какая характеристика переменной определяет множество значений, допустимых для нее, операции, выполняемые на этих значениях и количество выделяемой памяти?

Задание №8. Какое значение сохранится в переменной `length`, если на вход программы подать число 1024?

```
n = int(input())
length = 0
while n > 0:
    n //= 10
    length += 1
```

Задание №9. Дополните определение: «Доказательство правильности алгоритма называется».

Задание №10. Вставьте пропущенное слово в предложение « ... алгоритма характеризует логическую сложность и определяется количеством путей, по которым может реализоваться процесс вычислений».

Формируемая компетенция: ПК 1.4

Перечень заданий закрытого типа

Задание №1. Определите, что такое тестирование программного обеспечения?

- а) последовательность действий, которая осуществляется конечными пользователями непосредственно перед официальным выпуском программного обеспечения;
- б) направление на поиск отсутствующей или неверно работающей функциональности, ошибок в доступе к базе данных, ошибки инициализации, проблемы с производительностью, ошибки интерфейса;
- в) проверка соответствия программного обеспечения требованиям, осуществляемая с помощью наблюдения за его работой в специальных, искусственно построенных ситуациях.

Задание №2. Что является основной задачей тестирования?

- а) построить такой набор ситуаций, который был бы достаточно репрезентативен и позволял бы завершить тестирование с достаточной степенью уверенности в правильности программного обеспечения вообще и убедиться, что в конкретной ситуации оно работает правильно, в соответствии с требованиями;
- б) снижение вероятности наличия дефектов, находящихся в программном обеспечении.
- в) экономия времени команды тестировщиков, в случае, если релиз имеет серьезные проблемы со своей готовностью к полному циклу тестирования;

Задание №3. Что такое стратегия тестирования?

- а) когда подаются некоторые данные на вход и проверяются результаты, в надежде найти несоответствия;
- б) начало тестирования с терминальных классов (т.е. классов, не использующих методы других классов);
- в) система методов отбора и создания тестов для тестового набора.

Задание №4. Как называется попытка найти ошибки при выполнении программы в реальной среде?

- а) отладка;
- б) контроль;
- в) испытание;
- г) нет верного ответа.

Задание №5. Как называется процесс локализации и исправления ошибок, обнаруженных при тестировании ПО?

- а) локализацией;
- б) отладкой;
- в) инициализацией;
- г) нет верного ответа.

Задание №6. Как называется метод тестирования функционального поведения объекта с точки зрения внешнего мира?

- а) тестирование «белого ящика»;
- б) тестирование «серого ящика»;

в) «тестирование «черного ящика».

Задание №7. Сопоставьте между собой тип ошибок, выявленных при тестировании ПО, и характеристики ошибок.

| | |
|---|-------------------------|
| 1. Ошибки, обнаруженные компоновщиком при объединении модулей программы, называют | а) ошибками выполнения; |
| 2. Ошибки, фиксируемые компилятором при выполнении синтаксического и частично семантического анализа программы называют | б) ошибками компоновки; |
| 3. Ошибками кодирования называют | в) ошибками компиляции. |

Задание №8. Сопоставьте описание метода отладки программ и название метода.

| | |
|--|-----------------------------------|
| 1. Самый простой и естественный способ отладки программы. Метод эффективен, но не применим для программ со сложными вычислениями, для больших программ, а также в случаях, когда ошибка связана с неверным представлением программиста о выполнении операций | а) метод дедукции; |
| 2. Сначала формируют множество причин, которые могли бы вызвать данное проявление ошибки. Затем, анализируя причины, исключают те, которые противоречат имеющим данным | б) метод ручного тестирования; |
| 3. Метод основан на тщательном анализе симптомов ошибки, которые могут проявляться как неверные результаты вычислений или как сообщение об ошибке | в) метод обратного прослеживания; |
| 4. Начинается проверка с точки вывода неправильного результата. Для этой точки строится гипотеза о значениях основных переменных, которые могли бы привести к получению имеющегося результата | г) метод индукции. |

Задание №9. Установите правильную последовательность этапов процесса тестирования ПО.

- а) Формализация результатов и подготовка отчетов;
- б) Анализ требований;
- в) Разработка тестов;
- г) Выполнение тестирования;
- д) Планирование тестирования;
- е) Подготовка к тестированию.

Задание №10. Установите правильную последовательность содержания рекомендованного тестового сценария?

- а) Постусловие (Post Conditions);
- б) Описание (Summary);
- в) Предусловие (Pre Conditions);
- г) Ожидаемый результат (Expected Result);
- д) Шаги (степы) (Steps);
- е) Уникальное краткое название (ID);
- ж) Статус (Pass/Fail);

Перечень заданий открытого типа

Задание №1. Как называется набор входных значений, предусловий выполнения, ожидаемых результатов и постусловий выполнения, разработанный для определенной цели или тестового условия, таких как выполнение определенного пути программы, либо для проверки соответствия определенному требованию?

Задание № 2. Какой процесс содержит набор действий, которые планируют, проектируют, разрабатывают, производят, редактируют, распространяют и сопровождают те документы, в которых нуждаются все заинтересованные лица, такие как менеджеры, инженеры и пользователи программного средства?

Задание № 3. Для чего нужна спецификация тестирования программы?

Задание № 4. Какой программный документ содержит схему алгоритма, общее описание алгоритма и (или) функционирования программы, а также обоснование принятых технических и технико-экономических решений?

Задание № 5. Как называется граф, вершины которого кодируют операторы программы, а дуги - порядок исполнения операторов?

Задание № 6. К какой группе критериев тестирования ПО относятся критерий тестирования команд, критерий тестирования ветвей и критерий тестирования путей?

Задание № 7. Какой программный документ содержит назначение и область применения программы, технические, технико-экономические и специальные требования, предъявляемые к программе, необходимые стадии и сроки разработки, виды испытаний?

Задание № 8. Зачем нужен Log-файл?

Задание № 9. Укажите номер стандарта: «ГОСТ «Виды программ и программных документов» устанавливает виды программ и программных документов для вычислительных машин, комплексов и систем независимо от их назначения и области применения».

Задание № 10. Запишите название определения: «Определение соответствия разрабатываемого ПО ожиданиям и потребностям пользователя, требованиям к системе называется ... ».

Формируемая компетенция: ПК 1.5

Перечень заданий закрытого типа

Задание №1. Как называется процесс изменения внутренней структуры программного продукта, не затрагивающий её внешнего поведения и имеющий целью облегчение понимания программного кода?

- а) реконструкция;
- б) оптимизация;
- в) реструктуризация;
- г) рефакторинг.

Задание №2. Какое количество строк программного кода считается оптимальным размером для тела функции или класса?

- а) 35-45 строк;

- б) 20– 30 строк;
- в) 40-50 строк;
- г) 45-55 строк.

Задание №3. Выберите корректные имена переменных, используемых в программе:

- а) \$a_d, s4;
- б) B1, 5_Wes;
- в) totalScore; max_Weight;
- г) sum(), num.

Задание №4. Выберите корректные имена функций и методов, используемых в программе:

- а) Score(), Size (), Speed ();
- б) saveScore(), set_Size(), getSpeed();
- в) 1numbers(), Size(), Sum();
- г) Close_1(), 2logp1().

Задание №5. К какой категории относится проблема в сложной условной конструкции, где какая-то ветка никогда не исполняется из-за ошибки или изменения требований к программе?

- а) мёртвый код;
- б) дублирование;
- в) длинное имя функции;
- г) сложное условие.

Задание №6. Какие специальные программы используются для поиска узких мест программы?

- а) компиляторы;
- б) трансляторы
- в) профайлеры;
- г) интерпретаторы;

Задание №7. Сопоставьте названия проблем программного кода и их характеристики.

| | |
|--|--|
| 1. Переменная, параметр, метод или класс больше не используется: требования к программе изменились. | а) некорректность имен переменных, функций, классов; |
| 2. Один и тот же программный код выполняет одно и то же действие в нескольких местах программы. | б) мёртвый код; |
| 3. Имена должны сообщать, почему элемент программного кода существует, что он делает и как используется. | в) дублирование. |

Задание №8. Сопоставьте характеристики принципов оптимизации программного кода и их названия.

| | |
|---|------------------------|
| 1. Код должен легко поддаваться редактированию, интегрированию или удалению отдельных функций или возможности без необходимости вносить серьезные изменения в другие части программы. | а) производительность; |
| 2. Увеличение быстродействия программы минимум на 20-30% в сравнение с исходным вариантом. | б) время; |
| 3. Оптимизация и последующая отладка программы должны занимать небольшой период продолжительности. | в) естественность. |

Задание №9. Установите правильную последовательность этапов создания версии документа в системе контроля версий.

- а) ветвление;
- б) извлечение;
- в) обновление рабочей копии;
- г) слияние;
- д) фиксация изменений.

Задание №10. Установите правильную последовательность этапов рефакторинга программного кода.

- а) ревью и слияние с основной веткой;
- б) планирование;
- в) мониторинг и анализ;
- г) выполнение рефакторинга;
- д) тестирование после каждого изменения;
- е) обеспечение безопасности;
- ж) документирование изменений.

Перечень заданий открытого типа

Задание № 1. Как называется процесс модификации программы с целью повышения эффективности ее работы?

Задание № 2. Как называется метод оптимизации программы, означающий сохранение результата выполнения определенной функции, которое поможет избежать ее повторного выполнения?

Задание № 3. Как можно упростить предложенный фрагмент программного кода на Python, создающий список из строки, применив конструктор списков:

```
list = []
for char in 'abcdef':
    if char != 'c':
        list.append(char * 2)
print(list)
```

Задание № 4. Как называется метод временного хранения данных в памяти устройства пользователя?

Задание № 5. Как называется программное обеспечение для облегчения работы с изменяющейся информацией?

Задание № 6. Как называется центральное хранилище, которое содержит версии файлов и которое часто организуется средствами СУБД?

Задание № 7. В предложенном фрагменте программного кода выполните рефакторинг строки вывода результата.

```
def input_value (prompt):
    while True:
        value = int( input( prompt ) )
        if value < 0 :
            print ("Значение не должно быть отрицательным!")
        else :
```

return value

```
price = input_value ("Введите цену товара: ")
quantity = input_value ("Введите количество товара: ")
cost = price * quantity
print("Стоимость товара: ", cost)
```

Задание № 8. Какой пункт главного меню среды разработки приложений PyCharm содержит инструменты, которые помогают проводить рефакторинг программного кода?

Задание № 9. Запишите название определения: «Зафиксированное в репозитории (центральном хранилище файлов) состояние файла называется ».

Задание № 10. Запишите название определения: «Приведение в соответствие рабочих версий файлов с актуальными версиями в репозитории называется ».

5. КРИТЕРИИ ОЦЕНКИ

В ФГБОУ ВО «ДГТУ» для обучающихся по образовательным программам среднего профессионального образования применяется пятибалльная шкала знаний, умений, практического опыта.

Таблица 3

| Шкалы оценивания | | Критерии оценивания |
|---------------------------------|--------------|--|
| пятибалльная | зачет | |
| «Отлично» - 5 баллов | | <p>Показывает высокий уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует глубокое и прочное освоение материала; – исчерпывающе, четко, последовательно, грамотно и логически стройно излагает теоретический материал; – правильно формирует определения; – демонстрирует умения самостоятельной работы с нормативно-правовой литературой; – умеет делать выводы по излагаемому материалу. |
| «Хорошо» - 4 балла | | <p>Показывает достаточный уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует достаточно полное знание материала, основных теоретических положений; – достаточно последовательно, грамотно логически стройно излагает материал; – демонстрирует умения ориентироваться в нормативно-правовой литературе; – умеет делать достаточно обоснованные выводы по излагаемому материалу. |
| «Удовлетворительно» - 3 балла | Зачтено | <p>Показывает пороговый уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует общее знание изучаемого материала; – испытывает затруднения при ответах на дополнительные вопросы; – знает основную рекомендуемую литературу; – умеет строить ответ в соответствии со структурой излагаемого материала. |
| «Неудовлетворительно» - 2 балла | Не зачтено | <p>Ставится в случае:</p> <ul style="list-style-type: none"> – незнания значительной части программного материала; – не владения понятийным аппаратом дисциплины; – допущения существенных ошибок при изложении учебного материала; – неумения строить ответ в соответствии со структурой излагаемого вопроса; – неумения делать выводы по излагаемому материалу. |

Критерии оценки тестовых заданий

Таблица 4

| Процент выполненных тестовых заданий | Оценка |
|--------------------------------------|---------------------|
| до 50% | неудовлетворительно |
| 50-69% | удовлетворительно |
| 70-84% | хорошо |
| 85-100% | отлично |

КЛЮЧИ К ЗАДАНИЯМ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

Таблица 5

| Формируемые компетенции | № задания | Ответ | |
|-------------------------------|-------------------------------|---|---|
| ПК 1.1 | Задания закрытого типа | | |
| | № 1 | в | |
| | № 2 | г | |
| | № 3 | 1-г, 2-в, 3-б, 4-а | |
| | № 4 | 1-в, 2-г, 3-а, 4-б | |
| | № 5 | вжгаездб | |
| | Задания открытого типа | | |
| | № 1 | решение | |
| | № 2 | машина Тьюринга | |
| | № 3 | программа | |
| | № 4 | результат | |
| | № 5 | входных | |
| | ПК 1.4 | Задания закрытого типа | |
| | | № 1 | а |
| № 2 | | б | |
| № 3 | | 1-в, 2-а, 3-б | |
| № 4 | | 1-б, 2-в, 3-а | |
| № 5 | | евбдгжа | |
| Задания открытого типа | | | |
| № 1 | | тестирование | |
| № 2 | | метод «черного ящика» | |
| № 3 | | спецификация | |
| № 4 | | 19.101 | |
| № 5 | | валидацией | |
| ПК 1.5 | | Задания закрытого типа | |
| | | № 1 | в |
| | № 2 | б | |
| | № 3 | 1-б, 2-в, 3-а | |
| | № 4 | 1-б, 2-в, 3-а | |
| | № 5 | бвдаг | |
| | Задания открытого типа | | |
| | № 1 | рефакторинг | |
| | № 2 | list = [char * 2 for char in 'abcdef' if char != 'i'] print (list) | |
| | № 3 | print ("Стоимость товара: ", price * quantity) | |
| | № 4 | мёртвым | |
| | № 5 | Система контроля | |

**КЛЮЧИ К ЗАДАНИЯМ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ
КОМПЕТЕНЦИЙ**

Таблица 6

| Формируемые компетенции | № задания | Ответ |
|-------------------------|---|--------------------------------------|
| ПК 1.1 | Задания закрытого типа | |
| | № 1 | в |
| | № 2 | б |
| | № 3 | г |
| | № 4 | а |
| | № 5 | в |
| | № 6 | г |
| | № 7 | 1-г,2-в,3-б,4-а |
| | № 8 | 1-в,2-г,3-а,4-б |
| | № 9 | вжгаездб |
| | № 10 | ежгвдба |
| | Задания открытого типа | |
| | № 1 | предметной областью программирования |
| | № 2 | классами сложности |
| | № 3 | машина Тьюринга |
| | № 4 | полиномиальным |
| | № 5 | длительность решения |
| | № 6 | переменная |
| | № 7 | тип данных |
| | № 8 | 4 |
| № 9 | верификацией | |
| № 10 | разветвленность | |
| ПК 1.4 | Задания закрытого типа | |
| | № 1 | в |
| | № 2 | б |
| | № 3 | в |
| | № 4 | в |
| | № 5 | б |
| | № 6 | а |
| | № 7 | 1-в, 2-а, 3-б |
| | № 8 | 1-б, 2-а, 3-г, 4-в |
| | № 9 | бдвга |
| | № 10 | евбдгжа |
| | Задания открытого типа | |
| | № 1 | тестовый сценарий |
| | № 2 | документирование программы |
| | № 3 | для разработки тестового набора |
| | № 4 | пояснительная записка |
| | № 5 | управляющий граф программы |
| | № 6 | структурных критериев |
| | № 7 | техническое задание |
| № 8 | для фиксации результатов прогона test-suite | |
| № 9 | 19.101 | |

| | | |
|--------|-------------------------------|---|
| | № 10 | валидацией |
| ПК 1.5 | Задания закрытого типа | |
| | № 1 | г |
| | № 2 | б |
| | № 3 | в |
| | № 4 | б |
| | № 5 | а |
| | № 6 | в |
| | № 7 | 1-б, 2-в, 3-а |
| | № 8 | 1-в, 2-а, 3-б |
| | № 9 | бвдаг |
| | № 10 | бегджав |
| | Задания открытого типа | |
| | № 1 | оптимизация |
| | № 2 | меморизация |
| | № 3 | <code>list = [char * 2 for char in 'abcdef' if char != 'i'] print (list)</code> |
| | № 4 | кеширование |
| | № 5 | система контроля версиями |
| | № 6 | репозиторий |
| | № 7 | <code>print ("Стоимость товара: ", price * quantity)</code> |
| | № 8 | Refactor |
| № 9 | редакцией | |
| № 10 | синхронизацией | |

Критерии оценки тестовых заданий, заданий на дополнение, с развернутым ответом и на установление правильной последовательности

Верный ответ - 2 балла.

Неверный ответ или его отсутствие - 0 баллов.

Критерии оценки заданий на сопоставление

Верный ответ - 2 балла

1 ошибка - 1 балл

более 1-й ошибки или ответ отсутствует - 0 баллов.