

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лидинович
Должность: Ректор
Дата подписания: 27.03.2023 19:54:16
Уникальный программный ключ:
20b84ea6d19eae7c3c775fccd8365441470edec7

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

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПМ. 01 Разработка модулей программного обеспечения для
компьютерных систем

по УП. 01.01 УЧЕБНАЯ ПРАКТИКА

специальность	09.02.07 Информационные системы и программирование
квалификация	программист
уровень образования	СПО на базе основного общего образования/ среднего общего образования

Разработчик _____
подпись _____ Адеева М.Г., к.э.н., доцент

Фонд оценочных средств обсужден на заседании отделения ИТ

« 05 » 2023г., протокол № 1

Зав. отделением ИТ _____
подпись _____ Адеева М.Г., к.э.н., доцент

г. Каспийск - 2023

СОДЕРЖАНИЕ

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ	
.....	Ошибка! Закладка не определена.
2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ПРАКТИКИ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ.....	3
3. ОЦЕНКА ОСВОЕНИЯ УЧЕБНОЙ ПРАКТИКИ.....	5
3.1. Контроль и оценка освоения учебной практики по темам.....	5
4. ПЕРЕЧЕНЬ ЗАДАНИЙ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ....	6
5. КРИТЕРИИ ОЦЕНКИ.....	15

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы УП 01.01 Учебная практика и предназначен для контроля и оценки образовательных достижений, обучающихся (в т.ч. по самостоятельной работе студентов, далее – СРС), освоивших программу данного модуля.

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

Рабочей программой УП 01.01 Учебная практика предусмотрено формирование профессиональных компетенций:

1) ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;

2) ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием;

3) ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

Формой аттестации по практике УП 01.01 Учебная практика является зачет с оценкой.

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ПРАКТИКИ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

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

Таблица 1

Результаты обучения: знания, умения, практический опыт	Формируемые виды деятельности/компетенции
Знать:	ПМ. 01 Разработка модулей программного обеспечения для компьютерных систем/ ПК 1.1
31 – основные этапы разработки программного обеспечения;	
32 - основные принципы технологии структурного и объектно-ориентированного программирования;	
33 - актуальная нормативно-правовая база области документирования алгоритмов;	
Уметь:	
У1 - формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;	ПМ. 01 Разработка модулей программного обеспечения для компьютерных систем/ ПК 1.2
У2 - оформлять документацию на программные средства;	
У3 - оценка сложности алгоритма;	
Иметь практический опыт в:	
П1 - разрабатывать алгоритм решения поставленной задачи и реализации его средствами автоматизированного проектирования.	
Знать:	
31 – основные этапы разработки программного обеспечения;	
32 - основные принципы технологии структурного и объектно-ориентированного программирования;	
33 – знание API современных мобильных операционных	

Результаты обучения: знания, умения, практический опыт	Формируемые виды деятельности/компетенции
систем; Уметь: У1 - создавать программу по разработанному алгоритму как отдельный модуль; У2 – оформлять документацию на программные средства; У3 - осуществлять разработку кода программного модуля на языках низкого и высокого уровней в том числе для мобильных платформ; Иметь практический опыт в: П1 - разрабатывать код программного продукта на основе готовой спецификации на уровне модуля; П2 - разрабатывать мобильные приложения.	
Знать: З1 – основные принципы отладки и тестирования программных продуктов; З2 – инструментарий отладки программных продуктов; Уметь: У1 - выполнять отладку и тестирование программы на уровне модуля; У2 - оформлять документацию на программные средства; У3 - применять инструментальные средства отладки программного обеспечения; Иметь практический опыт в: П1 - использовать инструментальные средства на этапе отладки программного продукта; П2 - проводить тестирование программного модуля по определенному сценарию.	ПМ. 01 Разработка модулей программного обеспечения для компьютерных систем/ ПК 1.3

3. ОЦЕНКА ОСВОЕНИЯ УЧЕБНОЙ ПРАКТИКИ

3.1. Контроль и оценка освоения учебной практики по темам

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

Таблица 2

Элемент учебной практики	Формы и методы контроля			
	Текущий контроль		Промежуточная аттестация	
	Форма контроля	Проверяемые компетенции/знания/умения/ практический опыт	Форма контроля	Проверяемые компетенции/знания/умения/ практический опыт
3 семестр/5 семестр				
Тема 1. Разработка консольного приложения	Практическая работа	ПК 1.1, 31, 32, 33, У1, У2, У3, П1; ПК 1.2, 31, 32, У1, У2, У3, П1	Зачетная работа	ПК 1.1, 31, 32, 33, У1, У2, У3, П1; ПК 1.2, 31, 32, У1, У2, У3, П1
Тема 2. Разработка графического приложения	Практическая работа	ПК 1.1, 31, 32, 33, У1, У2, У3, П1; ПК 1.2, 31, 32, У1, У2, У3, П1		ПК 1.1, 31, 32, 33, У1, У2, У3, П1; ПК 1.2, 31, 32, У1, У2, У3, П1
Тема 3. Программирование на языке низкого уровня	Практическая работа	ПК 1.1, 31, 32, 33, У1, У2, У3, П1; ПК 1.2, 31, 32, У1, У2, У3, П1		ПК 1.1, 31, 32, 33, У1, У2, У3, П1; ПК 1.2, 31, 32, У1, У2, У3, П1
4 семестр/6 семестр				
Тема 4. Отладка и тестирование программного обеспечения.	Практическая работа	ПК 1.3, 31, 32, У1, У2, У3, П1, П2	Зачетная работа	ПК 1.3, 31, 32, У1, У2, У3, П1, П2
Тема 5. Документирование программного обеспечения.	Практическая работа	ПК 1.1, 33, У2; ПК 1.2, 31, У2; ПК 1.3, 31, У2; ПК 1.4, 31, У2		ПК 1.1, 33, У2; ПК 1.2, 31, У2; ПК 1.3, 31, У2; ПК 1.4, 31, У2
Тема 6. Основные платформы и языки разработки мобильных приложений	Практическая работа	ПК 1.2, 33, У1, У2, У3, П2		ПК 1.2, 33, У1, У2, У3, П2

4. ПЕРЕЧЕНЬ ЗАДАНИЙ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Формируемая компетенция ПК 1.1

Перечень заданий закрытого типа

Задание №1. Какая строка программного кода является началом функции в программе?

- а) for i in range(0, 12)::
- б) defHello(name)::
- в) if a == 0::
- г) print(name).

Задание №2. Как называется элемент программы, предназначенный для хранения данных в процессе выполнения программы?

- а) константа;
- б) метод;
- в) переменная;
- г) ячейка памяти.

Задание №3. Дана переменная a=12. Выберите правильный вариант вывода на консоль значения переменной a:

- а) print(“На улице ” + a + “ градусов”);
- б) print(“На улице ” + str(a) + “ градусов”);
- в) print(str (“На улице ” + a + “ градусов”));
- г) print(“На улице ” + int(a) + “ градусов”).

Задание №4. Как называются итерации, при разработке продукта по гибким методологиям?

- а) забеги;
- б) спринты;
- в) дейли;
- г) митапы.

Задание №5. Какая команда позволяет задать псевдоним при импорте библиотеки в Python?

- а) in;
- б) for;
- в) as;
- г) str.

Задание №6. На каком этапе разработки программного обеспечения выполняется формализованное описание задачи?

- а) постановка задачи;
- б) математическое описание задачи;
- в) разработка алгоритма;
- г) отладка и тестирование программы.

Задание №7. Сопоставьте команды и назначения команд языка программирования:

Назначение команды	Имя команды
1. Вывод в консоль	а) while

2. Цикл	б) input
3. Ввод информации	в) print()
4. Функция	г) if
5. Условие	д) def

Задание №8. Сопоставьте характеристики и названия принципов объектно-ориентированного программирования:

1. объединение данных с процедурами в единый программный блок. Процедуры и функции – методы.	а) Наследование;
2. новый объект может быть создан на основе ранее существующего объекта, при этом он сохраняет свойства и метод родного объекта, но может приобретать дополнительные свойства и заменять существующие.	б) Полиморфизм;
3. единое имя для одинаковых действий, которые могут выполняться с различными объектами.	в) Инкапсуляция.

Задание №9. Установите правильную последовательность шагов алгоритма перевода чисел из десятичной системы в двоичную систему счисления.

- Если же полученное частное больше 1, то мы заменяем исходное число на него и возвращаемся в пункт 2).
- Если число равно 0 или 1, то это и будет его двоичное представление.
- Если полученное частное равно 1, то его дописываем в первый разряд двоичного представления числа и прекращаем вычисления.
- Полученный остаток от деления записываем в последний разряд двоичного представления числа.
- Если число больше 1, то мы делим его на 2.

Задание №10. Упорядочите фазы этапа синтеза процесса компиляции программы:

- оптимизация машинно-независимого кода – повышение эффективности кода;
- оптимизация машинного кода – повышение эффективности целевого кода;
- генерация машинно-независимого кода – формирование кода на промежуточном языке, который не зависит от конечного компьютера;
- генерация машинного кода – формирование кода на машинном языке;
- распределение памяти – назначение каждой переменной адреса в одной из областей памяти.

Перечень заданий открытого типа

Задание №1. Как называется сборник подпрограмм или объектов, используемых для разработки программного обеспечения (ПО)?

Задание №2. Как называется модель для создания объектов определённого типа в объектно-ориентированном программировании?

Задание №3. Как называется функция или процедура, принадлежащая какому-то классу или объекту, состоящая из некоторого количества операторов для выполнения какого-то действия, может иметь набор входных аргументов?

Задание №4. Как называется сообщение, которое возникает в различных точках исполняемого кода при выполнении определённых условий?

Задание №5. Дополните определение: « ... - некоторая сущность в виртуальном пространстве, обладающая определённым состоянием и поведением, имеющая заданные значения свойств (атрибутов) и операций над ними».

Задание №6. Дополните строку программного кода на языке Python для создания объекта name по получению данных от пользователя: «... ('Ваше имя: ')».

Формируемая компетенция ПК 1.2

Перечень заданий закрытого типа

Задание №1. Сколько библиотек можно импортировать в один проект программы?

- а) не более 3;
- б) не более 10;
- в) не более 5;
- г) не более 23;
- д) неограниченное количество.

Задание №2. Как называется системная программа, которая преобразует символические конструкции в команды машинного языка (язык конкретного компьютера(процессора) или группы компьютеров (семейства процессоров))?

- а) транслятор;
- б) ассемблер
- в) перекодировщик;
- г) компилятор.

Задание №3. Как называется процесс трансляции с языковвысокого уровня программирования в программу на машинном языке?

- а) интерпретацией;
- б) эмуляцией;
- в) компиляцией;
- г) перекодированием.

Задание №4. Как называется процесс создания эффективного целевого программного кода?

- а) кодирование;
- б) оптимизация;
- в) эмуляция;
- г) компиляция.

Задание №5. Какие ошибки допущены в коде ниже?

```
def factorial(n):  
    if n == 0:  
        return 1  
    else  
        return n * factorial(n - 1)
```

```
print(factorial(5))
```

- а) функция не может вызывать сама себя;
- б) необходимо указать тип возвращаемого результата;
- в) не хватает двоеточия после конструкции else;

г) в коде нет никаких ошибок.

Задание №6. В какой строке кода происходит преобразование строки в целое число?

- а) `int(input('Ваш возраст?'));`
- б) `for(input('Ваш возраст?'));`
- в) `in(input('Ваш возраст?'));`
- г) `str(input('Ваш возраст?')).`

Задание №7. Сопоставьте тип переменной и назначение переменной адреса в одной из областей памяти:

1. время жизни переменной связано с временем жизни программы;	а) динамическая;
2. время жизни переменной связано с временем жизни программного блока;	б) глобальная;
3. время жизни переменной неизвестно.	в) статическая.

Задание №8. Сопоставьте названия и характеристики фаз этапа анализа процесса компиляции программы:

1. переход от последовательности знаков к символам языка, обработка пробелов и удаление комментариев в программном коде.	а) семантический анализ
2. определение общей структуры программы с пониманием порядка следования символов (грамматика).	б) лексический анализ
3. определение смыслового значения текста программы и проверка особых свойств (например, типов переменных и областей их видимости).	в) синтаксический анализ

Задание №9. Расставьте в правильной последовательности команды предложенной программы, которая определяет является введенное число четным или нечетным.

- а) `print('Нечетное');`
- б) `if n % 2 == 0;`
- в) `else;;`
- г) `n = int(input());`
- д) `print('Четное').`

Задание №10. Установите правильную последовательность этапов построения сложного системного ПО:

- а) научно-исследовательская работа;
- б) технология изготовления и испытания спроектированного объекта, внесения коррекции;
- в) техническое задание;
- г) технический проект;
- д) эскизный проект;
- е) рабочий проект;

Перечень заданий открытого типа

Задание №1. Как называется процесс поиска и устранения ошибок в программе.?

Задание №2. Оптимизируйте предложенный исходный программный код:

`a = b*c + b*d`
`e = e * 0`

Задание №3. Как называется бинарная форма представления кода определенного языка программирования?

Задание №4. Как называется версия программы, которая может быть доработана и изменена после получения обратной связи от пользователей, не готовая к финальному запуску?

Задание №5. Дополните предложенный программный код строкой подсчёта количества тёплых дней в месяце.

```
may_2021 = [24, 26, 15, 10, 15, 19, 10, 11, 14, 7, 7, 7, 12, 14, 17, 10, 13, 19, 21, 22, 11, 15, 19, 23, 15, 21, 16, 13, 25, 17, 19] # список ежедневных температур

def comfort_count(temperatures):
    count = 0
    for temp in temperatures:
        if 22 <= temp <= 26:
            ...
    return count
nice_days = comfort_count(may_2021)
print('Количество тёплых дней в этом месяце:', nice_days)
```

Задание №6. Дополните предложенный программный код строкой вывода на экран сообщения “У тебя {количество} друзей”, где {количество} - значение переменной count.

```
print("Привет, я ИИ")
friends = ['Сергей', 'Соня', 'Дима', 'Алина', 'Егор']
count = len(friends)
....
```

Формируемая компетенция ПК 1.3

Перечень заданий закрытого типа

Задание №1. Какая архитектура реализована во многих типах микропроцессоров, выполняющих большой набор разноформатных команд с использованием многочисленных способов адресации.

- а) RISC (Reduced Instruction Set Computer);
- б) CISC (Complex Instruction Set Computer);
- в) VLIW (Very Large Instruction Word);
- г) нет правильного ответа.

Задание №2. Какой тип тестирования оценивает способность программного продукта взаимодействовать с одним или более указанными компонентами ПО или системами?

- а) тестирование безопасности;
- б) тестирование производительности;
- в) тестирование удобства использования (usability);
- г) тестирование совместимости.

Задание №3. Как называют последовательности операторов программы, которые выполняются при конкретном варианте исходных данных?

- а) маршрутом;

- б) дорожкой;
- в) сектором;
- г) сегментом.

Задание №4. Как называется конкретный набор входных данных и ожидаемых результатов для тестирования функции?

- а) тестовый сценарий;
- б) тестовый кейс;
- в) тестовый случай;
- г) тестовый набор.

Задание №5. К какой группе процессов жизненного цикла ПО относят квалификационное тестирование программных средств?

- а) процессы организационного обеспечения проекта;
- б) процессы реализации программных средств;
- в) технические процессы;
- г) процессы поддержки программных средств.

Задание №6. Какая часть машинной команды содержит код, задающий вид операции (сложение, умножение, передача и т.д.)?

- а) операционная часть;
- б) адресная часть;
- в) форматная часть;
- г) структурная часть.

Задание №7. Сопоставьте классификационные признаки и соответствующие им виды тестирования ПО.

Классификационный признак	Виды тестирования	
1. По объекту тестирования	а)	компонентное тестирование; интеграционное тестирование.
2. По знанию системы	б)	ручное тестирование; автоматизированное тестирование.
3. По степени автоматизированности	в)	черный ящик; белый ящик.
4. По уровню тестирования	г)	функциональное тестирование; нагрузочное тестирование; тестирование производительности; тестирование удобства использования (usability); тестирование интерфейса пользователя; тестирование безопасности; тестирование совместимости.

Задание №8. Сопоставьте названия основных инструментальных средств отладки ПО и их определения.

1. Внутрисхемные эмуляторы;	а) программное средство, способное имитировать работу микроконтроллера и его памяти;
2. Программные симуляторы;	б) программно-аппаратное средство, способное замещать собой эмулируемый (моделируемый) процессор в реальной схеме;
3. Платы развития;	в) специальная программа, загружаемая в память отлаживаемой системы, вынуждает процессор

	пользователя производить, кроме прикладной задачи, еще и отладочные функции;
4. Мониторы отладки.	г) своеобразные конструкторы для макетирования прикладных систем.

Задание №9. Установите правильный порядок шагов базового рабочего процесса отладки программы на языке программирования Python.

- а) проверка значений;
- б) настройка точек останова;
- в) обработка исключений;
- г) пошаговое выполнение кода.

Задание №10. Определите правильную последовательность этапов процесса тестирования ПО.

- а) управление;
- б) планирование;
- в) создание отчетов о процессе тестирования;
- г) подготовку тестовых данных и выбор условий;
- д) разработку и выполнение тестовых сценариев;
- е) оценку критериев выхода;
- ж) проверку результатов.

Перечень заданий открытого типа

Задание №1. Как называется последовательность действий, которые выполняются для проверки определенной функции программы?

Задание №2. Как называется закодированное по определенным правилам указание микропроцессору на выполнение некоторой операции или действия?

Задание №3. Как называется адрес команды, при достижении которого выполнение программы должно быть приостановлено?

Задание №4. Как называется встроенный отладчик для программы на языке Python, который позволяет отлаживать программу в процессе её работы?

Задание №5. Дополните утверждение: «Первая и самая очевидная цель процесса тестирования – это выявление программы».

Задание №6. Дополните утверждение: «Информация, содержащаяся в дефектах программы, является базой для формирования представления об уровне ».

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		Показывает высокий уровень сформированности компетенций, т.е.: – 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		Показывает достаточный уровень сформированности компетенций, т.е.: – 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	Показывает пороговый уровень сформированности компетенций, т.е.: – 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	Ставится в случае: – 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

Таблица 5

Формируемые компетенции	№ задания	Ответ
ПК 1.1	Задания закрытого типа	
	№ 1	а
	№ 2	в
	№ 3	б
	№ 4	б
	№ 5	в
	№ 6	б
	№ 7	1-в, 2-а, 3-б, 4-д, 5-г
	№ 8	1-в, 2-а, 3-б
	№ 9	бдгва
	№ 10	вадбг
	Задания открытого типа	
	№ 1	библиотека
	№ 2	класс
	№ 3	метод
	№ 4	событие
	№ 5	Объект
	№ 6	name=input
ПК 1.2	Задания закрытого типа	
	№ 1	д
	№ 2	б
	№ 3	в
	№ 4	б
	№ 5	в
	№ 6	а
	№ 7	1-в, 2-а, 3-б
	№ 8	1-б, 2-в, 3-а
	№ 9	гбдва
	№ 10	вадгеб
	Задания открытого типа	
	№ 1	отладка
	№ 2	$a = b * (c + d)$ $e = 0$
	№ 3	двоичный код
	№ 4	бета-версия
	№ 5	count += 1

	№ 6	print("У тебя", count, "друзей")
ПК 1.3	Задания закрытого типа	
	№ 1	б
	№ 2	г
	№ 3	а
	№ 4	в
	№ 5	б
	№ 6	а
	№ 7	1-г, 2-в, 3-б, 4-а
	№ 8	1-б, 2-а, 3-г, 4-в
	№ 9	бгав
	№ 10	багджев
	Задания открытого типа	
	№ 1	тестовый сценарий
	№ 2	машинная команда
	№ 3	точка останова
	№ 4	pdb
	№ 5	дефектов
	№ 6	качества программы

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.