

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Ладиевич
Должность: Ректор
Дата подписания: 07.06.2024 09:18
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

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

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине ОПЦ.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

специальность

09.02.07 Информационные системы и
программирование

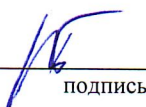
квалификация

программист

уровень образования

СПО на базе основного общего образования/
среднего общего образования

Разработчик


подпись

Адиева М.Г., к.э.н., доцент

Фонд оценочных средств обсужден на заседании отделения ИТ

« 25 » 09 2023г., протокол № 1

Зав. отделением ИТ


подпись

Адиева М.Г., к.э.н., доцент

г. Махачкала - 2023

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..	11
5. КРИТЕРИИ ОЦЕНКИ.....	19

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств (далее ФОС) является неотъемлемой частью рабочей программы дисциплины ОПЦ.04 «Основы алгоритмизации и программирования» и предназначен для контроля и оценки образовательных достижений, обучающихся (в т.ч. по самостоятельной работе студентов), освоивших программу данной дисциплины.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям федеральных государственных образовательных стандартов среднего профессионального образования (далее - ФГОС СПО) по специальности 09.02.07 Информационные системы и программирование.

Рабочей программой дисциплины ОПЦ.04 «Основы алгоритмизации и программирования» предусмотрено формирование следующих 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.

Формой аттестации по учебной дисциплине является экзамен.

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

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

Таблица 1

Результаты обучения: знания, умения, практический опыт	Формируемые виды деятельности/ компетенции
Знать:	ПК 1.1
З1 - основные этапы разработки программного обеспечения;	
З2 - основные принципы технологии структурного и объектно-ориентированного программирования	
Уметь:	
У1 - формировать алгоритмы разработки программных модулей в соответствии с техническим заданием	
Иметь практический опыт:	ПК 1.2
П1 - разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования	
Знать:	
З1 - основные этапы разработки программного обеспечения;	
З2 - основные принципы технологии структурного и объектно-ориентированного программирования.	
Уметь:	ПК 1.2
У1 - создавать программу по разработанному алгоритму как отдельный модуль	

Результаты обучения: знания, умения, практический опыт	Формируемые виды деятельности/ компетенции
У2 - осуществлять разработку кода программного модуля на языках низкого и высокого уровней, в том числе для мобильных платформ.	
Иметь практический опыт:	
П1 - разрабатывать код программного продукта на основе готовой спецификации на уровне модуля	

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. Введение в дисциплину ОАиП. Алгоритм и его свойства.	Устный опрос; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1
Тема 2. Основные структуры алгоритмов	Устный опрос; Лабораторная работа № 1; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1
Тема 3. Логические основы алгоритмизации	Устный опрос; Лабораторная работа № 2; Лабораторная работа № 3; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1
Тема 4. Языки и системы программирования	Письменная работа; Устный опрос; Лабораторная работа № 4; Круглый стол; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1
Тема 5. Методы и принципы	Устный опрос; Круглый стол;	ПК 1.1, 31, 32, У1, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1

программирования.	Самостоятельная работа			
Тема 6. Основы языка программирования Python. Переменные и типы данных	Устный опрос; Лабораторная работа № 5; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 7. Арифметические и логические операции.	Устный опрос; Лабораторная работа № 6; Лабораторная работа № 7; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 8. Условные операторы	Письменная работа; Устный опрос; Лабораторная работа № 8; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 9. Циклические операторы	Устный опрос; Лабораторная работа № 9; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 10. Строки. Методы строк.	Устный опрос; Лабораторная работа № 10; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 11. Списки. Кортежи.	Устный опрос; Лабораторная работа № 11; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 12. Функции	Устный опрос; Лабораторная работа № 12; Самостоятельная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1
Тема 13. Введение в ООП. Классы	Письменная работа; Устный опрос; Лабораторная работа № 13; Лабораторная работа № 14.	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1	Экзаменационная работа	ПК 1.1, 31, 32, У1, П1; ПК 1.2, 31, 32, У1, У2, П1

3.2. Перечень заданий для текущего контроля

Формируемая компетенция: ПК 1.1

Перечень заданий закрытого типа

Задание №1. Как называется команда:

**Если условие
То действия при выполнении условия
Всё**

на псевдокоде?

- а) линейным алгоритмом;
- б) циклом;
- в) неполным ветвлением;
- г) полным ветвлением.

Задание №2. Какого типа данных является переменная $A = -12,65$?

- а) int;
- б) double;
- в) string;
- г) bool.

Задание №3. Сопоставьте принципы ООП и их краткие характеристики.

1. Инкапсуляция;	а)касается способности языка позволять строить новые определения классов на основе определений существующих классов.
2. Наследование;	б) это свойство, которое позволяет одно и тоже имя использовать для решения нескольких технически разных задач.
3. Полиморфизм	в)это механизм программирования, объединяющий вместе код и данные, которыми он манипулирует, исключая как вмешательство извне, так и неправильное использование данных.

Задание №4. Сопоставьте какая алгоритмическая конструкция подходит для каждой из приведенных постановок задач?

1. Вычислить произведение и сумму чисел от 10 до 50.	а) линейная;
2. Вычислить значение выражения для $x=8$ и $y=16$: $z = 1 + \frac{x^2}{2} + \frac{y^3}{4}$	б) циклическая;
3. Даны два действительных числа. Заменить первое число нулем, если оно меньше второго или равно ему, и оставить числа без изменения в противном случае.	в) разветвляющаяся.

Задание №5. Установите правильную последовательность этапов решения поставленной задачи на ЭВМ.

- а) построение алгоритма;
- б) формализация задачи;
- в) постановка задачи;
- г) составление программы на языке программирования;
- д) проведение расчетов и анализ полученных результатов;
- е) отладка и тестирование программы.

Перечень заданий открытого типа

Задание №1. На каком этапе разработки программного обеспечения устанавливаются состав и формы представления входной, промежуточной и результатной информации?

Задание №2. Как называется набор инструкций, описывающих порядок действий исполнителя для достижения некоторого результата?

Задание №3. Каким блочным символом можно изобразить в блок-схеме следующую строку программного кода: `for i in range(1, num + 1, 2):` ?

Задание №4. Дополните предложение: «Поименованная ячейка памяти носит название ...».

Задание №5. Дополните предложение словосочетанием: «Блок ... используется для указания обращений к вспомогательным алгоритмам, существующим автономно в виде некоторых самостоятельных модулей, и для обращений к библиотечным подпрограммам».

Формируемая компетенция: ПК 1.2

Перечень заданий закрытого типа

Задание №1. Определите, что будет результатом выполнения арифметического выражения: `9%2`?

- а) 4;
- б) 1;
- в) 4.5;
- г) 81.

Задание №2. В языке Python циклом с предусловием является:

- а) `do-while`;
- б) `iterate`;
- в) `while`;
- г) `for`.

Задание №3. Сопоставьте названия методов для списков в стандартной библиотеке Python и их назначения.

1. <code>append()</code>	а) считает, сколько раз элемент повторяется в списке.
2. <code>extend()</code>	б) переставляет элементы в обратном порядке.
3. <code>count()</code>	в) добавляет новый элемент в конец списка.
4. <code>reverse()</code>	г) добавляет набор элементов в конец списка.

Задание №4. Сопоставьте результаты выполнения программ с предложенными программными решениями.

Номер	Программное решение
1.	<pre>num = 5 factorial = 1 if num < 0: print("Для отрицательных чисел, факториал не существует") elif num == 0: print("Факториал 0 равен 1") else: for i in range(1, num + 1):</pre>

	<pre>factorial = factorial * i print(factorial)</pre>
2.	<pre>input_number = 6 print ('Куб числа равен', input_number**3)</pre>
3.	<pre>k = 3 m = 5 n = 9 if n <= k: t = 2 * m elif n * 2 % k == 0: t = m * (n * 2 // k) else: t = m * (1 + (n * 2 // k)) print(t)</pre>

Результаты выполнения программ:

- а) 216;
- б) 30;
- в) 120;

Задание №5. Расположите номера строк предложенного программного кода решения поставленной задачи в правильной последовательности.

Постановка задачи: «Выведите на экран все числа, кратные определенному числу, в заданном диапазоне».

Номер строки кода	Строки программного кода
1	<pre>if (i % n == 0):</pre>
2	<pre>lower = int(input("Введите нижнюю границу диапазона: "))</pre>
3	<pre>print(i)</pre>
4	<pre>n = int(input("Введите делитель: "))</pre>
5	<pre>upper = int(input("Введите верхнюю границу диапазона: "))</pre>
6	<pre>for i in range(lower, upper + 1):</pre>

Перечень заданий открытого типа

Задание №1. Какая результатная информация отобразится на экране после выполнения программы, если на вход программы подать a=20, b=15?

```
a = int(input())
b = int(input())
if a < b:
    print(a)
else:
    print(b)
```

Задание №2. Укажите номер строки программы, в которой может возникнуть ошибка?
Гарантируется, что на вход программы подаются только целые числа.

```
1 n = int(input())
2 k = int(input())
3 for i in range(n, k):
4     if i % 10 == 0:
5         t += 1
6     else:
7         t = 0
```

Задание №3. Укажите, что будет выведено на экран при выполнении программы?

```
a = 168
b = 72
while a != b:
    if a > b:
        a -= b
    else:
        b -= a
print(a)
```

Задание №4. Дополните предложение: «Встроенная функция Python ... преобразует значение в число с плавающей точкой».

Задание №5. Дополните третью строку структуры объявления функции оператором, после которого указывается возвращаемое значение.

```
1 def имя_функции ([параметры]):
2     инструкции
3     .... возвращаемое значение
```

4. ПЕРЕЧЕНЬ ЗАДАНИЙ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Формируемая компетенция: ПК 1.1

Перечень заданий закрытого типа

Задание №1. На каком этапе разработки программного обеспечения устанавливаются состав и формы представления входной, промежуточной и результатной информации?

- а) математическое описание;
- б) постановка задачи;
- в) составление программы;
- г) тестирование и отладка программы;
- д) опытная эксплуатация.

Задание № 2. Выберите из предложенных выражений формулировку принципа структурного программирования:

- а) сложная задача разбивается на достаточно простые части, каждая из которых имеет один вход и один выход;
- б) моделирование требуемых атрибутов и взаимодействий сущностей выполняется в виде классов для определения абстрактного представления системы;
- в) создание новых абстракций возможно на основе существующих;
- г) внутреннее состояние и функции объекта скрыты и предоставление доступа осуществляется только через открытый набор функций.

Задание № 3. Выберите из предложенных выражений формулировку принципа объектно-ориентированного программирования:

- а) сложная задача разбивается на достаточно простые части, каждая из которых имеет один вход и один выход;
- б) алгоритм и программа должны составляться поэтапно (по шагам);
- в) создание новых абстракций возможно на основе существующих;
- г) логика алгоритма и программы должна опираться на минимальное число достаточно простых базовых управляющих структур.

Задание № 4. Как называется алгоритмическая конструкция, обеспечивающая выбор между двумя альтернативами в зависимости от значения входных данных?

- а) разветвляющейся;
- б) циклической;
- в) линейной;
- г) комбинированной.

Задание № 5. Как называется принцип моделирования требуемых атрибутов и взаимодействий сущностей в виде классов для определения основного представления системы?

- а) инкапсуляция.
- б) наследование.
- в) абстракция.
- г) полиморфизм.

Задание № 6. Что из нижеперечисленного записывается внутри данного блочного символа

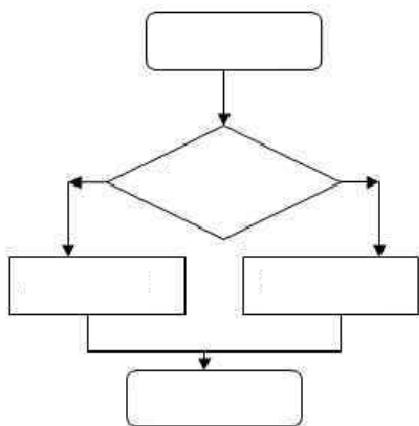


при составлении алгоритма решения задачи?

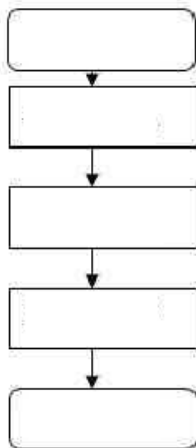
- а) ввод входных данных;
- б) комментарий;
- в) условие;
- г) вычислительное действие;
- д) начальное значение, граничное условие и шаг изменения значения параметра для повторения.

Задание № 7. Сопоставьте названия алгоритмических конструкций с изображениями рисунков?

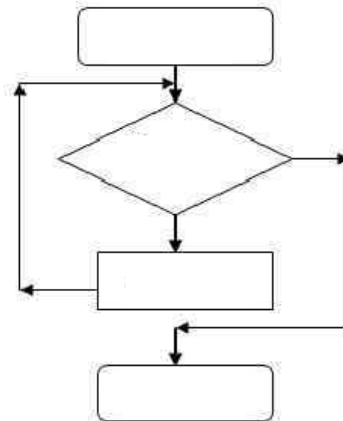
1



2



3



- а) линейная;
- б) циклическая;
- в) разветвляющаяся.

Задание № 8. Сопоставьте какая алгоритмическая конструкция подходит для каждой из приведенных постановок задач?

1

Вычислить произведение и сумму чисел от 10 до 50.

2

Вычислить значение выражения для $x=8$ и $y=16$:

$$z = 1 + \frac{x^2}{2} + \frac{y^3}{4}$$

3

Даны два действительных числа. Заменить первое число нулем, если оно меньше второго или равно ему, и оставить числа без изменения в противном случае.

- а) линейная;
- б) циклическая;
- в) разветвляющаяся.

Задание № 9. Установите правильную последовательность шагов алгоритма решения квадратного уравнения.

- а) Если $(D > 0)$, вычислить $X1 = (-B + \sqrt{D})/(2*A)$, $X2 = (-B - \sqrt{D})/(2*A)$, вывести сообщение "Два корня, X1 = ", X1, " X2 = ", X2.
- б) Вычислить дискриминант $D = B^2 - 4*A*C$.

- в) Если ($D < 0$), вывести сообщение "Уравнение действительных решений не имеет".
- г) Если ($D = 0$), вычислить $X = -B/(2*A)$, вывести сообщение "Один корень, $X =$ ", X .
- д) Получить коэффициенты уравнения - A, B, C .

Задание № 10. Установите правильную последовательность этапов решения поставленной задачи на ЭВМ.

- а) построение алгоритма;
- б) формализация задачи;
- в) постановка задачи;
- г) составление программы на языке программирования;
- д) проведение расчетов и анализ полученных результатов;
- е) отладка и тестирование программы.

Перечень заданий открытого типа

Задание №1. На каком этапе разработки программного обеспечения выбирается форма хранения данных и составляются все необходимые формулы?

Задание №2. На какие виды делятся данные по отношению к программе?

Задание №3. Какой принцип объектно-ориентированного программирования формулируется как скрытие внутреннего состояния и функций объекта и предоставление доступа только через открытый набор функций?

Задание №4. Как называется набор инструкций, описывающих порядок действий исполнителя для достижения некоторого результата?

Задание №5. Как называется способ описания структуры алгоритма с помощью последовательности связанных между собой функциональных блоков?

Задание № 6. Как называется свойство алгоритма, определяемое как разбиение алгоритма на ряд отдельных законченных действий?

Задание №7. Как называется отдельная функционально законченная программная единица, которая структурно оформляется стандартным образом по отношению к компилятору и по отношению к объединению ее с другими аналогичными единицами в загрузке?

Задание №8. Каким блочным символом можно изобразить в блок-схеме следующую строку программного кода: `for i in range(1, num + 1, 2):` ?

Задание №9. Описание структуры алгоритма на естественном, частично формализованном языке, позволяющее выявить основные этапы решения задачи, перед точной его записью на языке программирования называется ...?

Задание №10. Совокупность действий, направленных на устранение ошибок в программах, начиная с момента обнаружения фактов ошибочной работы программы и завершая устранением причин их возникновения называется ...?

Формируемая компетенция: ПК 1.2

Перечень заданий закрытого типа

Задание № 1. Какой метод позволяет получить данные от пользователя?

- а) read();
- б) get();
- в) cin();
- г) input().

Задание № 2. Какая функция выводит информацию в консоль (на экран монитора)?

- а) out();
- б) log();
- в) print();
- г) write();

Задание № 3. Какие существуют типы переменных у чисел в Python (выбрать несколько вариантов):

- а) float;
- б) list;
- в) num;
- г) int;
- д) cout;
- е) str.

Задание № 4. Какая результатная информация отобразится на экране после выполнения программы:

```
main.py
1 a=5
2 b=a+5
3 a=b*100
4 print(a)
```

- а) 2500;
- б) 1000;
- в) 25;
- г) 100.

Задание № 5. Какая результатная информация отобразится на экране после выполнения программы, если на вход программы подать a=10, b=20?

```
a = int(input())
b = int(input())
if a < b:
    print(a)
else:
    print(b)
```

- а) 10;
- б) 20;
- в) 30;
- г) -10.

Задание № 6. Укажите, значение переменной **i** после выполнения данного фрагмента программы:

```
i=1

while i<=10:
    i=i+1
```

- а) 9;
- б) 10;
- в) 11;
- г) 2.

Задание № 7. Сопоставьте постановки задач и предложенные программные решения.

1)	2)	3)
<pre>n = int(input()) s = 0 for i in range(1,n + 1, 1): s += i print(s)</pre>	<pre>num = 75869 count = 0 while num != 0: num = num // 10 count = count + 1 print("Чисел:", count)</pre>	<pre>start = 25 end = 50 for num in range(start, end + 1): if num > 1: for i in range(2, num): if (num % i) == 0: break else: print(num)</pre>

- а) Программа, которая считает количество чисел в цифре;
- б) Программа, которая выводит все простые числа, из определенного диапазона;
- в) Программа, которая вычисляет сумму всех чисел от 1 до заданного пользователем числа;

Задание № 8. Сопоставьте результаты выполнения программ с предложенными программными решениями.

Номер	Программное решение
1.	<pre>num = 5 factorial = 1 if num < 0: print("Для отрицательных чисел, факториал не существует") elif num == 0: print("Факториал 0 равен 1") else: for i in range(1, num + 1): factorial = factorial * i print(factorial)</pre>
2.	<pre>input_number = 6 print ('Куб числа равен', input_number**3)</pre>
3.	<pre>k = 3 m = 5 n = 9 if n <= k: t = 2 * m elif n * 2 % k == 0:</pre>

<pre> t = m * (n * 2 // k) else: t = m * (1 + (n * 2 // k)) print(t) </pre>

Результаты выполнения программ:

- а) 216;
- б) 30;
- в) 120;

Задание № 9. Расположите номера строк предложенного программного кода решения поставленной задачи в правильной последовательности.

Постановка задачи: «Задан год в виде натурального числа. Нужно выяснить, является ли год с данным номером високосным или нет. Если год високосный, то напечатайте "YES", а если нет, то выведите NO. Год является високосным, если он кратен 400 или его номер кратен 4, но не кратен 100».

Номер строки кода	Строки программного кода
1	else:
2	if a % 4 == 0 and a % 100 != 0:
3	a = int(input())
4	print('YES')
5	print('NO')
6	print('YES')
7	elif a % 400 == 0:

Задание № 10. Расположите номера строк предложенного программного кода решения поставленной задачи в правильной последовательности.

Постановка задачи: «Выведите на экран все числа, кратные определенному числу, в заданном диапазоне».

Номер строки кода	Строки программного кода
1	if (i % n == 0):
2	lower = int(input("Введите нижнюю границу диапазона: "))
3	print(i)
4	n = int(input("Введите делитель: "))
5	upper = int(input("Введите верхнюю границу диапазона: "))
6	for i in range(lower, upper + 1):

Перечень заданий открытого типа

Задание № 1. Укажите, сколько раз выполнится тело цикла с данным заголовком:


```
i=1
while i<=10:
    i=i+1
```

Задание № 2. Что будет выведено на экран монитора в результате выполнения программы:

```
main.py
1  z=5
2  z1=z**3
3  print(z1)
4
```

Задание № 3. Укажите номер строки программы, в которой может возникнуть ошибка? Гарантируется, что на вход программы подаются только целые числа.

```
1  n = int(input())
2  k = int(input())
3  for i in range(n, k):
4      if i % 10 == 0:
5          t += 1
6      else:
7          t = 0
```

Задание № 4. Укажите, что будет выведено на экран при выполнении программы?

```
a = 168
b = 72
while a != b:
    if a > b:
        a -= b
    else:
        b -= a
print(a)
```

Задание № 5. Укажите, что будет выведено на экран при выполнении программы?

```

a = 1
b = 1
rez = 2
for i in range(2, 20, 3):
    a, b = b, a + b
    rez += 2 * b
print(rez)

```

Задание № 6. Напишите функцию `perimetr`, вычисляющую периметр прямоугольника со сторонами `a` и `b`.

Задание № 7. Напишите функцию `isEven`, возвращающую `True`, если число четное, и `False`, если - нечетное.

Задание № 8. Что происходит в системе программирования при создании объекта с помощью класса?

Задание № 9. Дан построчный фрагмент программного кода, который принимает три положительных числа и определяет вид треугольника, длины сторон которого равны введенным числам. Дополните программу недостающим оператором в строке кода с номером 1.

№ строки	Программный код
1	
2	<code>if a1 == a2 == a3:</code>
3	<code>print('Равносторонний')</code>
4	<code>elif a2 == a3 or a1 == a3 or a1 == a2:</code>
5	<code>print('Равнобедренный')</code>
6	<code>else:</code>
7	<code>print('Разносторонний')</code>

Задание № 10. Дан фрагмент программного кода, который принимает целое число `x` и определяет, принадлежит ли данное число промежутку от -1 до 17. Дополните оператор `if` в строке с номером 2 необходимым условием.

№ строки	Программный код
1	<code>a1 = int(input())</code>
2	<code>if _____:</code>
3	<code>print('Принадлежит')</code>
4	<code>else:</code>
5	<code>print('Не принадлежит')</code>

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	б	
	№ 3	1-в, 2-а, 3-б	
	№ 4	1-б, 2-а, 3-в	
	№ 5	вбагед	
	Задания открытого типа		
	№ 1	постановка задачи	
	№ 2	алгоритм	
	№ 3	модификацией	
	№ 4	переменной	
	№ 5	предопределенный процесс	
	ПК 1.2	Задания закрытого типа	
		№ 1	б
		№ 2	в
№ 3		1-в, 2-г, 3-а, 4-б	
№ 4		1-в, 2-а, 3-б	
№ 5		516324	
Задания открытого типа			
№ 1		15	
№ 2		5	
№ 3		24	
№ 4		float()	
№ 5		return	

КЛЮЧИ К ЗАДАНИЯМ ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Таблица 6

Формируемые компетенции	№ задания	Ответ
ПК 1.1	Задания закрытого типа	
	№ 1	б
	№ 2	а
	№ 3	в
	№ 4	а

	№ 5	в
	№ 6	д
	№ 7	1-в, 2-а, 3-б
	№ 8	1-б, 2-а, 3-в
	№ 9	дбвга
	№ 10	вбагед
	Задания открытого типа	
	№ 1	формализации
	№ 2	исходные, результаты и промежуточные
	№ 3	инкапсуляция
	№ 4	алгоритм
	№ 5	графический
	№ 6	дискретность
	№ 7	модуль
	№ 8	модификацией
	№ 9	псевдокодом
	№ 10	отладкой
ПК 1.2	Задания закрытого типа	
	№ 1	г
	№ 2	в
	№ 3	а, г, е
	№ 4	б
	№ 5	а
	№ 6	в
	№ 7	1-в, 2-а, 3-б
	№ 8	1-в, 2-а, 3-б
	№ 9	6213754
	№ 10	516324
	Задания открытого типа	
	№ 1	10
	№ 2	125
	№ 3	5
	№ 4	24
	№ 5	106
	№ 6	def perimetr(a, b): return 2 * (a + b) p = perimetr(4, 3) print(p)
	№ 7	def isEven(x): return x % 2 == 0 print(isEven(10)) print(isEven(11))
	№ 8	запускается конструктор выбранного класса, и атрибутам нового объекта передаются значения аргументов
№ 9	a1, a2, a3= int(input()), int(input()), int(input())	
№ 10	a1 > -1 and a1 < 17	

**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.