

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гаранин Максим Алексеевич
Должность: Ректор
Дата подписания: 22.01.2023 09:45:18
Уникальный программный ключ:
7708e3a47e66a8ee02711b298d7c78bd1e40bf88

Приложение
к рабочей программе дисциплины

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

**ОСНОВЫ АЛГОРИТМИЗАЦИИ И
ПРОГРАММИРОВАНИЕ**

(наименование дисциплины(модуля))

Направление подготовки / специальность

13.03.02 Электроэнергетика и электротехника

(код и наименование)

Направленность (профиль)/специализация

Электрический транспорт

(наименование)

Содержание

1. Пояснительная записка.
2. Типовые контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций.
3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации.

1. Пояснительная записка

Цель промежуточной аттестации – оценивание промежуточных и окончательных результатов обучения по дисциплине, обеспечивающих достижение планируемых результатов освоения образовательной программы.

Формы промежуточной аттестации: Зачёт, 7 семестр

Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции	Код индикатора достижения компетенции
ОПК-2: Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-2.1: Применяет современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения

Результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ОПК-2.4: Решает задачи в области профессиональной деятельности, используя перспективные методы машинного обучения	Обучающийся знает : основные конструкции языка программирования высокого уровня; основные типы данных и операторы; алгоритмы решения типовых задач с использованием алгоритмов ветвления; алгоритмы решения типовых задач с использованием циклических алгоритмов;	Вопросы (№1 - №30)
	Обучающийся умеет : писать программы на языке программирования высокого уровня; писать программы с использованием подпрограмм; писать программы для анализа информации на веб-страницах.	Задания (№1 - №30)
	Обучающийся владеет : разработки алгоритмов и написания программ, пригодных для автоматизации работы с веб-данными;	Задания (№ 31- №33)

Промежуточная аттестация (экзамен) проводится в одной из следующих форм:

- 1) ответ на билет, состоящий из теоретических вопросов и практических заданий;
- 2) выполнение заданий в ЭИОС СамГУПС.

Промежуточная аттестация (зачет) проводится в одной из следующих форм:

- 1) собеседование;
- 2) выполнение заданий в ЭИОС СамГУПС.

2. Типовые¹ контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций

2.1. Типовые вопросы (тестовые задания) для оценки знаниевого образовательного результата

Проверяемый образовательный результат:

¹ Приводятся типовые вопросы и задания. Оценочные средства, предназначенные для проведения аттестационного мероприятия, хранятся на кафедре в достаточном для проведения оценочных процедур количестве вариантов. Оценочные средства подлежат актуализации с учетом развития науки, образования, культуры, экономики, техники, технологий и социальной сферы. Ответственность за нераспространение содержания оценочных средств среди обучающихся университета несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-2.1: Применяет современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения	Обучающийся знает: основные конструкции языка программирования высокого уровня; основные типы данных и операторы; алгоритмы решения типовых задач с использованием алгоритмов ветвления; алгоритмы решения типовых задач с использованием циклических алгоритмов;
<p>A1. Какие из нижеперечисленных парадигм программирования поддерживает язык Python</p> <p>a) объектно-ориентированное; b) структурное; c) функциональное; d) все вышеперечисленные?</p> <p>A2. Язык Python является:</p> <p>a) регистрозависимым; b) компилируемым; c) высокоуровневым; d) всё вышеперечисленное.</p> <p>A3. Инструкция list задает</p> <p>a) кортеж; b) список; c) множество; d) строку.</p> <p>A4. Инструкция tuple задает</p> <p>a) кортеж; b) список; c) множество; d) строку.</p> <p>A5. Равенство в Python обозначается конструкцией</p> <p>a) <> b) == c) != d) =!</p> <p>A6. Что будет являться результатом выполнения следующих строк кода x = input ('введите число') print(type(x))</p> <p>a) <class 'int';> b) <class 'float';> c) <class 'str';> d) класс переменной будет зависеть от введенных данных?</p> <p>A7. Какая библиотека используется для работы с регулярными выражениями в Python?</p> <p>a) request; b) requests; c) re; d) random?</p> <p>A8. С помощью какой инструкции в языке Python определяются функции</p> <p>a) function; b) procedure; c) def; d) proc?</p> <p>A9. Функция open ('text.txt', 'r') открывает файл text.txt...</p> <p>a) На запись;</p>	

- b) На чтение;
- c) На дозапись;
- d) На чтение и запись.

A10. Что будет результатом выполнения следующих строк кода

```
b=6  
a=b=8  
print(a)
```

- a) 6;
- b) 8;
- c) False;
- d) True?

A11. Чему будет равна переменная sum результате выполнения следующего участка кода

```
sum = 0  
for i in range(1, n + 1):  
    sum += i
```

- a) n!;
- b) (n+1)!;
- c) n;
- d) n+1?

A12. S='программа'. Результатом выполнения команды list(S) будет

- a) ['п', 'р', 'о', 'г', 'р', 'а', 'м',,];
- b) ['п', 'р', 'о', 'г', 'р', 'а', 'м', 'м', 'а'];
- c) ['программа'];
- d) возникнет ошибка.

A13. S='5'. Каким будет результат выполнения команды S*3

- a) 15;
- b) '15';
- c) '555';
- d) возникнет ошибка?

A14. Какая из следующих конструкций вернёт длину слова s

- a) s.length;
- b) s.length();
- c) s.len;
- d) len(s)?

A15. S='программа'. Каким будет результат выполнения команды S[::-1]

- a) ['п', 'р', 'о', 'г', 'р', 'а', 'м', 'м', 'а']
- b) 'аммаргорп'
- c) 'программ'
- d) 'рограмма'?

A16. Что будет выведено на экран в результате выполнения следующего участка кода

```
a=6.7  
b = a // 2  
pt=rint(a)
```

- a) 3;
- b) 3.35;
- c) Syntax Error;
- d) 0.7?

A17. Что будет выведено на экран в результате выполнения следующего участка кода

```
a = [1,2, None,(),[],]  
print(len(a))
```

- a) 3;
- b) 5;
- c) Syntax Error;
- d) Infinity?

A18. Что будет выведено на экран в результате выполнения следующего участка кода

```
x=[1,2,3,4,5,6]
```

```
Print(x[3])
```

- a) 3;
- b) 4;
- c) Syntax Error;
- d) [1,2,3]?

A19. Что будет выведено на экран в результате выполнения следующего участка кода

```
x = [1, 2, 7]
```

```
y = x
```

```
y[-1] = ['a', 'b']
```

```
print x
```

- a) [1, 2, 'a', 'b'];
- b) [1, 2, ['a', 'b']];
- c) ['a', 'b', 1, 2];
- d) [[1, 2], 'a', 'b']?

A20. Что будет выведено на экран в результате выполнения следующего участка кода?

```
a=6
```

```
while a<10:
```

```
    print(str(a)*3,end='')
```

```
    a+=2
```

- a) 1824;
- b) 18 24;
- c) 666888;
- d) 666 888.

Оценочный лист к типовому заданию А (модельный ответ):

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
d	d	b	a	b	d	c	c	b	b
A11	A12	A13	A14	A15	A16	A17	A18	A19	A20
a	b	c	d	b	a	b	b	b	c

2.2. Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-2.1: Применяет современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения	Обучающийся умеет: писать программы на языке программирования высокого уровня; писать программы с использованием подпрограмм; писать программы для анализа информации на веб-страницах.
<p>V1. Что будет выведено на экран в результате выполнения следующего участка кода, если на вход были поданы числа 15,16,17? Приведите пример входных данных, при которых на экран будет выведено число 3.</p> <pre>a,b,c=(int(input()) for i in range(3)) print(max(a,b,c)%4)</pre> <p>V2. Ниже записаны две рекурсивные функции (процедуры): F и G. Сколько символов «звёздочка» будет напечатано на экране при выполнении вызова F(11)?</p> <pre>Python def F(n):</pre>	<p>V1. Что будет выведено на экран в результате выполнения следующего участка кода, если на вход были поданы числа 15,16,17? Приведите пример входных данных, при которых на экран будет выведено число 3.</p>

```

if n > 0: G(n - 1)
def G(n):
    print("*")
    if n > 1: F(n - 3)

```

В3. На обработку поступает положительное целое число, не превышающее 10^9 . Нужно написать программу, которая выводит на экран сумму цифр этого числа, меньших 7. Если в числе нет цифр, меньших 7, требуется на экран вывести 0. Программист написал программу неправильно.

```

N = int(input())
sum = 0
while N > 0:
    digit = N % 10
    if digit < 7:
        sum = sum + 1
    N = N // 10
print(digit)

```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 358.
2. Приведите пример такого трёхзначного числа, при вводе которого программа выдаёт верный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки: 1) выпишите строку, в которой сделана ошибка; 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Оценочный лист к типовому заданию В

В1. 1; 0,0,3

Программа выдаёт остаток от целочисленного деления максимального из чисел a, b и c на 4. В примере это 17 и результат целочисленного деления - 1. Подходящим ответом на вторую часть задания будет любой набор чисел, в котором максимальное число при делении на 4 в остатке даёт 3.

В2. 3

В3. Программа работает неправильно из-за неверной выводимой на экран переменной и неверного увеличения суммы. Соответственно, программа будет работать верно, если в числе старшая цифра (крайняя левая) равна сумме цифр, меньших 7.

1. Программа выведет число 3.
2. Пример числа, при вводе которого программа выдаёт верный ответ: 862.
3. В программе есть две ошибки.
 - 1) Неверное увеличение суммы. Строка с ошибкой: `sum = sum + 1`; Верное исправление: `sum = sum + digit`.
 - 2) Неверный вывод ответа на экран. Строка с ошибкой: `print(digit)` Верное исправление: `print(sum)`.

```

a,b,c=(int(input()))
for i in range(3))
    print(max(a,b,c)%4)

```

В2. Ниже записаны две рекурсивные функции (процедуры): F и G. Сколько символов «звёздочка» будет напечатано на экране при выполнении вызова F(11)?

```

Python def F(n):
    if n > 0: G(n - 1)
def G(n):
    print("*")
    if n > 1: F(n - 3)

```

В3. На обработку поступает положительное целое число, не превышающее 10^9 . Нужно написать программу, которая выводит на экран сумму цифр этого числа, меньших 7. Если в числе нет цифр, меньших 7, требуется на экран вывести 0. Программист написал программу неправильно.

```

N = int(input())
sum = 0
while N > 0:
    digit = N % 10
    if digit < 7:
        sum =
        sum + 1
    N = N // 10
print(digit)

```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 358.
2. Приведите пример такого трёхзначного числа, при вводе которого программа выдаёт верный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки: 1) выпишите строку, в которой

сделана ошибка; 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Оценочный лист к
типовому заданию В

В1. 1; 0,0,3

Программа выдаёт остаток от целочисленного деления максимального из числе a,b и c на 4. В примере это 17 и результат целочисленного деления - 1. Подходящим ответом на вторую часть задания будет любой набор чисел, в котором максимальное число при делении на 4 в остатке дает 3.

В2. 3

В3. Программа работает неправильно из-за неверной выводимой на экран переменной и неверного увеличения суммы. Соответственно, программа будет работать верно, если в числе старшая цифра (крайняя левая) равна сумме цифр, меньших 7.

1. Программа выведет число 3.

2. Пример числа, при вводе которого программа выдаёт верный ответ: 862.

3. В программе есть две ошибки.

1) Неверное увеличение суммы. Строка с ошибкой: `sum = sum + 1;`
Верное исправление: `sum = sum + digit.`

2) Неверный вывод ответа на экран. Строка с ошибкой: `print(digit)`
Верное исправление: `print(sum).`

ОПК-2.1: Применяет современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения

Обучающийся владеет:

Навыками разработки алгоритмов и написания программ, пригодных для автоматизации работы с веб-данными;

1. Напишите программу, которая в автоматическом режиме читает текст раздела «Руководство. Педагогический (научно-педагогический) состав» со страницы <https://www.samgups.ru/sveden/> и выводит его на экран.

2. Напишите программу, которая в автоматическом режиме выводит все направления подготовки, реализуемые в СамГУПС (Наименование специальности, направления подготовки со страницы <https://www.samgups.ru/sveden/education/study>)
3. Напишите программу, которая в автоматическом режиме выводит суммарную численность студентов очного отделения СамГУПС, обучающихся за счёт бюджетных ассигнований федерального бюджета, используйте данные со страницы «Образование» (<https://www.samgups.ru/sveden/education/study>)
4. Напишите программу, которая в автоматическом режиме выводит названия всех образовательных программ, реализуемых в СамГУПС («Образовательная программа») со страницы <https://www.samgups.ru/sveden/education/eduAccred>)
5. Напишите программу, которая в автоматическом режиме выводит суммарную численность обучающихся, являющихся иностранными гражданами, используйте данные со страницы «Образование» (<https://www.samgups.ru/sveden/education/study>)

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

1. Типы данных. Числа.
2. Типы данных. Строки. Методы работы со строками.
3. Типы данных. Списки. Методы работы со списками.
4. Индексы и срезы.
5. Глубокие и поверхностные копии.
6. Типы данных. Множества. Методы работы с множествами.
7. Типы данных. Словари.
8. Операторы ввода.
9. Логический тип данных.
10. Условий оператор. Конструкция if elif else
11. Оператор цикла for
12. Оператор цикла while
13. Инструкция break
14. Инструкция continue.
15. Именные функции. Инструкции def и return.
16. Lambda-функции.
17. Библиотека request
18. Регулярные выражения
19. Библиотека Beautiful soup.
20. Этапы парсинга сайтов.

3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации

Критерии формирования оценок по ответам на вопросы, выполнению тестовых заданий

- оценка **«отлично»** выставляется обучающемуся, если количество правильных ответов на вопросы составляет 100 – 90% от общего объёма заданных вопросов;
- оценка **«хорошо»** выставляется обучающемуся, если количество правильных ответов на вопросы – 89 – 76% от общего объёма заданных вопросов;
- оценка **«удовлетворительно»** выставляется обучающемуся, если количество правильных ответов на тестовые вопросы – 75–60 % от общего объёма заданных вопросов;
- оценка **«неудовлетворительно»** выставляется обучающемуся, если количество правильных ответов – менее 60% от общего объёма заданных вопросов.

Критерии формирования оценок по результатам выполнения заданий

- «Отлично»** – ставится за работу, выполненную полностью без ошибок и недочетов.
- «Хорошо»** – ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов.
- «Удовлетворительно»** – ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и двух недочетов.
- «Неудовлетворительно»** – ставится за работу, если число ошибок и недочетов превысило норму

для оценки «удовлетворительно» или правильно выполнено менее 2/3 всей работы.

Виды ошибок:

- *грубые ошибки: незнание основных понятий, правил, норм; незнание приемов решения задач; ошибки, показывающие неправильное понимание условия предложенного задания.*
- *негрубые ошибки: неточности формулировок, определений; нерациональный выбор хода решения.*
- *недочеты: нерациональные приемы выполнения задания; отдельные погрешности в формулировке выводов; небрежное выполнение задания.*

Критерии формирования оценок по зачету

«Зачтено» – студент демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности.

«Незачтено» – выставляется в том случае, когда студент демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.