ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА»

Кафедра «Вычислительной и прикладной математики»

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ**

**«Учебная практика»**

Направление подготовки

09.03.03 «Прикладная информатика»

Направленность (профиль) подготовки

Прикладная информатика

Квалификация выпускника – бакалавр

Форма обучения – очная, заочная

Рязань

**1. ОБЩИЕ ПОЛОЖЕНИЯ**

Оценочные материалы – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной профессиональной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям ОПОП.

Основная задача – обеспечить оценку уровня сформированности общекультурных, общепрофессиональных и профессиональных компетенций.

Контроль знаний обучающихся проводится в форме промежуточной аттестации.

Промежуточный контроль по дисциплине осуществляется проведением зачета с оценкой.

**2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ**

Сформированность каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

1. пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;
2. продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
3. эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

**Уровень освоения компетенций, формируемых дисциплиной:**

*а) описание критериев и шкалы оценивания тестирования:*

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| 3 балла(эталонный уровень) | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 85 до 100% |
| 2 балла(продвинутый уровень) | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 75 до 84% |
| 1 балл(пороговый уровень) | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 60 до 74% |
| 0 баллов | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 0 до 59% |

*б) описание критериев и шкалы оценивания теоретического вопроса:*

|  |  |
| --- | --- |
| ***Шкала оценивания*** | **Критерий** |
| 3 балла(эталонный уровень) | выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя. |
| 2 балла(продвинутый уровень) | выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов. |
| 1 балл(пороговый уровень) | выставляется студенту, который дал неполный ответ на вопрос в билете и смог ответить на дополнительные вопросы только с помощью преподавателя.  |
| 0 баллов | выставляется студенту, который не смог ответить на вопрос  |

*в) описание критериев и шкалы оценивания практического задания:*

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| 3 балла(эталонный уровень) | Задача решена верно |
| 2 балла(продвинутый уровень) | Задача решена верно, но имеются технические неточности в расчетах |
| 1 балл(пороговый уровень) | Задача решена верно, с дополнительными наводящими вопросами преподавателя |
| 0 баллов | Задача не решена |

***На зачет с оценкой***выносится: тестовое задание, 1 практическое задание и 1 теоретический вопрос. Студент может набрать максимум 9 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| отлично(эталонный уровень) | 8 – 9 баллов | Обязательным условием является выполнение всех предусмотренных в течение семестра заданий  |
| хорошо(продвинутый уровень) | 6 – 7 баллов |
| удовлетворительно(пороговый уровень) | 4 – 5 баллов |
| неудовлетворительно | 0 – 3 баллов | Студент не выполнил всех предусмотренных в течение семестра текущих заданий  |

3. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Контролируемые разделы (темы) дисциплины**(результаты по разделам) | **Код контролируемой компетенции (или её части)** | **Наименование оценочного мероприятия** |
|
| 1 | 2 | 3 | 4 |
| 1 | Тема 1. Среда программирования и структура программы на языке Питон | УК-1.1 ОПК-2.1 ОПК-2.2ОПК-7.1 ОПК-7.2 ОПК-7.3 | Зачет с оценкой |
| 2 | Тема 2. Программирование разветвляющихся алгоритмов | УК-1.1 ОПК-2.1 ОПК-2.2ОПК-7.1 ОПК-7.2 ОПК-7.3 | Зачет с оценкой |
| 3 | Тема 3. Программирование циклических алгоритмов | УК-1.1 ОПК-2.1 ОПК-2.2ОПК-7.1 ОПК-7.2 ОПК-7.3 | Зачет с оценкой |
| 4 | Тема 4. Программирование практических задач | УК-1.1 ОПК-2.1 ОПК-2.2ОПК-7.1 ОПК-7.2 ОПК-7.3 | Зачет с оценкой |

**4. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ**

 *4.1. Промежуточная аттестация (зачет с оценкой)*

|  |
| --- |
| **УК-1: Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач** |
| **УК-1.1. Осуществляет поиск необходимой информации, подвергает ее критическому анализу и обобщению** |

|  |
| --- |
| **ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решения задач профессиональной деятельности;** |
| **ОПК-2.1. Понимает состояние и тенденции развития современных информационных технологий и программных средств, в том числе отечественного производства** |
| **ОПК-2.2. Использует при решении задач профессиональной деятельности современные информационные технологии и программные средства, в том числе отечественного производства** |
| **ОПК-7: Способен разрабатывать алгоритмы и программы, пригодные для практического использования**  |
| **ОПК-7.1. Понимает требования к алгоритмам, суть процесса алгоритмизации задач**  |
| **ОПК-7.2. Выполняет разработку алгоритмического и программного обеспечения для решения прикладных задач.** |
| **ОПК-7.3. Применяет основы информатики и программирования к проекту, конструирует и тестирует программный продукт** |

**Вопросы к зачету по учебной практике**

[1. Что такое Python?](https://dev-ed.ru/blog/101-python3-qa/#1)

[2. В каком году написана первая статья про Python?](https://dev-ed.ru/blog/101-python3-qa/#2)

[3. Какие типы данных есть в Python? На какие классы делятся?](https://dev-ed.ru/blog/101-python3-qa/#3)

[4. Что такое лямбда-функция? Какое у неё назначение?](https://dev-ed.ru/blog/101-python3-qa/#4)

[5. Что такое PEP 8?](https://dev-ed.ru/blog/101-python3-qa/#5)

[6. Как получить документацию по атрибутам объекта?](https://dev-ed.ru/blog/101-python3-qa/#6)

[7. Что такое docstring?](https://dev-ed.ru/blog/101-python3-qa/#7)

[8. В чём разница между типами list и tuple?](https://dev-ed.ru/blog/101-python3-qa/#8)

[9. Может ли быть индекс списка отрицательным?](https://dev-ed.ru/blog/101-python3-qa/#9)

[10. Что значит конструкция pass?](https://dev-ed.ru/blog/101-python3-qa/#10)

[11. Чем отличаются многопоточное и многопроцессорное приложение?](https://dev-ed.ru/blog/101-python3-qa/#11)

[12. Как просмотреть методы объекта?](https://dev-ed.ru/blog/101-python3-qa/#12)

[13. Что такое \*args и \*\*kwargs в определении функции?](https://dev-ed.ru/blog/101-python3-qa/#13)

[14. Python полностью поддерживает ООП?](https://dev-ed.ru/blog/101-python3-qa/#14)

[15. Что такое globals() и locals()?](https://dev-ed.ru/blog/101-python3-qa/#15)

[16. Что хранится в атрибуте \_\_dict\_\_?](https://dev-ed.ru/blog/101-python3-qa/#16)

[17. Как проверить файл .py на синтаксические ошибки, не запуская его?](https://dev-ed.ru/blog/101-python3-qa/#17)

[18. Зачем в Python используется ключевое слово self?](https://dev-ed.ru/blog/101-python3-qa/#18)

[19. Что такое декоратор? Как написать собственный?](https://dev-ed.ru/blog/101-python3-qa/#19)

[20. Что может быть ключом в словаре?](https://dev-ed.ru/blog/101-python3-qa/#20)

[21. В чём разница между пакетами и модулями?](https://dev-ed.ru/blog/101-python3-qa/#21)

[22. Как перевести строку, содержащую двоичный код (1 и 0), в число?](https://dev-ed.ru/blog/101-python3-qa/#22)

[23. Для чего используется функция \_\_init\_\_?](https://dev-ed.ru/blog/101-python3-qa/#23)

[24. Что такое слайс(slice)?](https://dev-ed.ru/blog/101-python3-qa/#24)

[25. Как проверить, что один кортеж содержит все элементы другого кортежа?](https://dev-ed.ru/blog/101-python3-qa/#25)

[26. Почему пустой список нельзя использовать как аргумент по умолчанию?](https://dev-ed.ru/blog/101-python3-qa/#26)

[27. Что такое @classmethod, @staticmethod, @property?](https://dev-ed.ru/blog/101-python3-qa/#27)

[28. Что такое синхронный код?](https://dev-ed.ru/blog/101-python3-qa/#28)

[29. Что такое асинхронный код? Приведите пример.](https://dev-ed.ru/blog/101-python3-qa/#29)

[30. Каким будет результат следующего выражения?](https://dev-ed.ru/blog/101-python3-qa/#30)

-30 % 10

[31. Для чего нужен метод id()?](https://dev-ed.ru/blog/101-python3-qa/#31)

[32. Что такое итератор?](https://dev-ed.ru/blog/101-python3-qa/#32)

[33. Что такое генератор? Чем отличается от итератора?](https://dev-ed.ru/blog/101-python3-qa/#33)

[34. Для чего используется ключевое слово yield?](https://dev-ed.ru/blog/101-python3-qa/#34)

[35. Чем отличаются \_\_iter\_\_ и \_\_next\_\_?](https://dev-ed.ru/blog/101-python3-qa/#35)

[36. Что такое контекстный менеджер?](https://dev-ed.ru/blog/101-python3-qa/#36)

[37. Как сделать Python-скрипт исполняемым в различных операционных системах?](https://dev-ed.ru/blog/101-python3-qa/#37)

[38. Как сделать копию объекта? Как сделать глубокую копию объекта?](https://dev-ed.ru/blog/101-python3-qa/#38)

[39. Опишите принцип работы сборщика мусора в Python](https://dev-ed.ru/blog/101-python3-qa/#39)

[40. Как использовать глобальные переменные? Это хорошая идея?](https://dev-ed.ru/blog/101-python3-qa/#40)

[41. Для чего в классе используется атрибут \_\_slots\_\_?](https://dev-ed.ru/blog/101-python3-qa/#41)

[42. Какие пространства имен существуют в Python?](https://dev-ed.ru/blog/101-python3-qa/#42)

[43. Как реализуется управление памятью в Python?](https://dev-ed.ru/blog/101-python3-qa/#43)

[44. Что такое метаклассы и в каких случаях их следует использовать?](https://dev-ed.ru/blog/101-python3-qa/#44)

[45. Зачем нужен pdb?](https://dev-ed.ru/blog/101-python3-qa/#45)

[46. Каким будет результат следующего выражения?](https://dev-ed.ru/blog/101-python3-qa/#46)

[0, 1][10:]

[47. Как создать класс без слова class?](https://dev-ed.ru/blog/101-python3-qa/#47)

[48. Как перезагрузить импортированный модуль?](https://dev-ed.ru/blog/101-python3-qa/#48)

[49. Напишите декоратор, который будет перехватывать ошибки и повторять функцию максимум N раз.](https://dev-ed.ru/blog/101-python3-qa/#49)

[50. Каким будет результат следующего выражения?](https://dev-ed.ru/blog/101-python3-qa/#50)

len(' '.join(list(map(str, [[0], [1]]))))

[51. Python — легкий язык. Согласны?](https://dev-ed.ru/blog/101-python3-qa/#51)

[52. Какие проблемы есть в Python?](https://dev-ed.ru/blog/101-python3-qa/#52)

[53. Когда будет выполнена ветка else в конструкции try...except...else?](https://dev-ed.ru/blog/101-python3-qa/#53)

[54. Поддерживает ли Python множественное наследование?](https://dev-ed.ru/blog/101-python3-qa/#54)

[55. Как dict и set реализованы внутри? Какова сложность получения элемента? Сколько памяти потребляет каждая структура?](https://dev-ed.ru/blog/101-python3-qa/#55)

[56. Что такое MRO? Как это работает?](https://dev-ed.ru/blog/101-python3-qa/#56)

[57. Как аргументы передаются в функции: по значению или по ссылке?](https://dev-ed.ru/blog/101-python3-qa/#57)

[58. С помощью каких инструментов можно выполнить статический анализ кода?](https://dev-ed.ru/blog/101-python3-qa/#58)

[59. Что будет напечатано в результате выполнения следующего кода?](https://dev-ed.ru/blog/101-python3-qa/#59)

import sys

arr\_1 = []

arr\_2 = arr\_1

print(sys.getrefcount(arr\_1))

[60. Что такое GIL? Почему GIL всё ещё существует?](https://dev-ed.ru/blog/101-python3-qa/#60)

[61. Опишите процесс компиляции в Python.](https://dev-ed.ru/blog/101-python3-qa/#61)

[62. Как тиражировать Python код?](https://dev-ed.ru/blog/101-python3-qa/#62)

[63. Что такое дескрипторы? Есть ли разница между дескриптором и декоратором?](https://dev-ed.ru/blog/101-python3-qa/#63)

[64. Почему всякий раз, когда Python завершает работу, не освобождается вся память?](https://dev-ed.ru/blog/101-python3-qa/#64)

[65. Что будет напечатано в результате выполнения следующего кода?](https://dev-ed.ru/blog/101-python3-qa/#65)

class Variable:

 def \_\_init\_\_(self, name, value):

 self.\_name = name

 self.\_value = value

 @property

 def value(self):

 print(self.\_name, 'GET', self.\_value)

 return self.\_value

 @value.setter

 def value(self, value):

 print(self.\_name, 'SET', self.\_value)

 self.\_value = value

var\_1 = Variable('var\_1', 'val\_1')

var\_2 = Variable('var\_2', 'val\_2')

var\_1.value, var\_2.value = var\_2.value, var\_1.value

[66. Что такое интернирование строк? Почему это есть в Python?](https://dev-ed.ru/blog/101-python3-qa/#66)

[67. Как упаковать бинарные зависимости?](https://dev-ed.ru/blog/101-python3-qa/#67)

[68. Почему в Python нет оптимизации хвостовой рекурсии? Как это реализовать?](https://dev-ed.ru/blog/101-python3-qa/#68)

[69. Что такое wheels и eggs? В чём разница?](https://dev-ed.ru/blog/101-python3-qa/#69)

[70. Как получить доступ к модулю, написанному на Python из C и наоборот?](https://dev-ed.ru/blog/101-python3-qa/#70)

[71. Как ускорить существующий код Python?](https://dev-ed.ru/blog/101-python3-qa/#71)

[72. Что такое \_\_pycache\_\_? Что такое файлы .pyc?](https://dev-ed.ru/blog/101-python3-qa/#72)

[73. Что такое виртуальное окружение?](https://dev-ed.ru/blog/101-python3-qa/#73)

[74. Python — это императивный или декларативный язык?](https://dev-ed.ru/blog/101-python3-qa/#74)

[75. Что такое менеджер пакетов? Какие менеджеры пакетов вы знаете?](https://dev-ed.ru/blog/101-python3-qa/#75)

[76. В чём преимущества массивов NumPy по сравнению с (вложенными) списками Python?](https://dev-ed.ru/blog/101-python3-qa/#76)

[77. Вам нужно реализовать функцию, которая должна использовать статическую переменную. Вы не можете писать код вне функции и у вас нет информации о внешних переменных (вне вашей функции). Как это сделать?](https://dev-ed.ru/blog/101-python3-qa/#77)

[78. Что будет напечатано в результате выполнения следующего кода?](https://dev-ed.ru/blog/101-python3-qa/#78)

def f\_g():

 yield 43

 return 66

print(f\_g())

[79. Как имплементировать словарь с нуля?](https://dev-ed.ru/blog/101-python3-qa/#79)

[80. Напишите однострочник, который будет подсчитывать количество заглавных букв в файле.](https://dev-ed.ru/blog/101-python3-qa/#80)

[81. Что такое файлы .pth?](https://dev-ed.ru/blog/101-python3-qa/#81)

[82. Какие функции из Collections и Itertools вы используете?](https://dev-ed.ru/blog/101-python3-qa/#82)

[83. Что делает флаг PYTHONOPTIMIZE?](https://dev-ed.ru/blog/101-python3-qa/#83)

[84. Что будет напечатано в результате выполнения следующего кода?](https://dev-ed.ru/blog/101-python3-qa/#84)

arr = [[]] \* 5

arr\_1, arr\_2 = arr, arr

for k, arr in enumerate((arr\_1, arr\_2)):

 arr[0].append(k)

arr = (arr\_1, 5, arr\_2)

print(arr)

[85. Какие переменные среды, влияющие на поведение интерпретатора Python, вы знаете?](https://dev-ed.ru/blog/101-python3-qa/#85)

[86. Что такое Cython? Что такое IronPython? Что такое PyPy? Почему они до сих пор существуют и зачем?](https://dev-ed.ru/blog/101-python3-qa/#86)

[87. Как перевернуть генератор?](https://dev-ed.ru/blog/101-python3-qa/#87)

[88. Приведите пример использования filter и reduce над итерируемым объектом.](https://dev-ed.ru/blog/101-python3-qa/#88)

[89. Что будет напечатано в результате выполнения кода?](https://dev-ed.ru/blog/101-python3-qa/#89)

print(\_)

[90. Чем фреймворк отличается от библиотеки?](https://dev-ed.ru/blog/101-python3-qa/#90)

[91. Расположите функции в порядке эффективности, объясните выбор.](https://dev-ed.ru/blog/101-python3-qa/#91)