Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финуниверситет)
Калужский филиал Финуниверситета
Кафедра «Бизнес-информатика и высшая математика»

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПОДГОТОВКЕ, НАПИСАНИЮ И ОФОРМЛЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ № 1 ПО ДИСЦИПЛИНЕ «Алгоритмы и структуры данных в языке Python» 1 семестр

Для студентов, обучающихся по направлению подготовки
38.03.05 –Бизнес-информатика
Образовательная программа «Цифровая трансформация управления бизнесом»

КАЛУГА 2026

Методические рекомендации по подготовке, написанию и оформлению контрольной работы по дисциплине «Алгоритмы и структуры данных в языке Python» 1 семестр предназначены для студентов, обучающихся по направлению подготовки 38.03.05 – Бизнес-информатика образовательная программа «Цифровая трансформация управления бизнесом» по очной форме обучения.

Составитель Костенко А. В. доцент, канд. пед. н., доцент
(Ф.И.О., должность, ученая степень и звание)

Рекомендовано Учебно-методическим советом Калужского филиала Финуниверситета
(протокол № 01 от 28 августа 2026 г.)

Одобрено кафедрой «Бизнес-информатика и высшая математика» Калужского филиала Финуниверситета
(протокол № 01 от 28 августа 2026 г.)

СОДЕРЖАНИЕ

Общие положения	4
Порядок выполнения контрольной работы	4
Требования к выполнению контрольной работы	5
Критерии оценки контрольной работы	5
Структура контрольной работы	6
Выбор варианта контрольной работы	7
Варианты контрольной работы	7
Планируемые результаты освоения	12
Требования к оформлению контрольной работы	13

ОБЩИЕ ПОЛОЖЕНИЯ

Контрольная работа является одной из форм аудиторной и внеаудиторной самостоятельной работы студентов и может реализовываться как в письменном виде так и с использованием информационных технологий и специализированных программных продуктов.

Контрольная работа отражает степень освоения студентами учебного материала конкретных разделов (тем) дисциплин и оформляется в форме развернутых ответов на вопросы, раскрытия понятий, выполнения упражнений, решения практических задач, ситуаций, кейсов и др.

Цель выполнения контрольной работы, содержащей комплект заданий – овладение студентами навыками решения типовых расчетных задач, формирование учебно-исследовательских навыков, закрепление умений самостоятельно работать с различными источниками информации; проверка сформированности компетенций.

Содержание заданий контрольных работ должно охватывать основной материал соответствующих разделов (тем) дисциплин. Контрольные задания разрабатываются по многовариантной системе. Варианты контрольных работ должны быть равноценны по объему и сложности.

Содержание заданий контрольных работ и требования к их выполнению разрабатываются преподавателем, ведущим семинарские (практические) занятия по дисциплине.

Оценка контрольных работ студентов проводится в процессе текущего контроля успеваемости студентов.

ПОРЯДОК ВЫПОЛНЕНИЯ КОНТРОЛЬНОЙ РАБОТЫ

Контрольная работа выполняется обучающимся в соответствии с заданием и методическими рекомендациями. Написанию контрольной работы должно предшествовать изучение указанных в тематике контрольных работ нормативных правовых актов и других источников, анализ и усвоение содержащихся в них положений.

При изложении теоретических вопросов не допускается простое переписывание источников. Ответы на вопросы обучающийся должен излагать самостоятельно. Решение практических заданий должно сопровождаться соответствующей аргументацией.

Сроки представления контрольной работы на проверку определяются календарным учебным графиком и приказом «Об организации учебного процесса на соответствующий учебный год».

Не допускается предъявление контрольной работы на проверку во время экзамена (зачета).

Выполненную контрольную работу обучающийся сдает для регистрации на кафедру, где она регистрируется в соответствующем

журнале. Данный журнал заводится на каждый курс/группу, структурируется по изучаемым на данном курсе учебным дисциплинам, по которым предусмотрено выполнение письменных работ.

Зарегистрированные в журнале учета работ студента контрольные работы получает под роспись преподаватель кафедры, за которым закреплена учебная нагрузка в части проверки контрольной работы по соответствующей учебной дисциплине.

В журнале фиксируется дата получения работы от обучающегося, дата получения контрольной работы преподавателем (заверяется подписью преподавателя), результат проверки контрольной работы, дата окончания проверки работы преподавателем (заверяется подписью преподавателя).

Контрольная работа должна быть проверена преподавателем в течение 10 рабочих дней с момента её получения у старшего лаборанта кафедры.

По результатам проверки контрольной работы выставляется оценка «зачтено» или «не зачтено» на титульном листе контрольной работы, заносится в «Ведомость учета проверенных работ».

Контрольные работы, по которым выставлена оценка «зачтено», обучающимся не возвращаются и хранятся на кафедре до конца учебного года, после чего уничтожаются. В случае отсутствия возможности хранения работ на кафедре, работы передаются в архив филиала на срок хранения, предусмотренный номенклатурой дел.

Не зачтенная контрольная работа (кроме работ, содержащих материалы ограниченного доступа) возвращается обучающемуся вместе с указаниями преподавателя по устранению недостатков, для повторного выполнения контрольной работы.

На титульном листе повторно выполненной контрольной работы старший лаборант кафедры делает пометку «повторно» и передает для проверки преподавателю кафедры.

ТРЕБОВАНИЯ К ВЫПОЛНЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ

- четкость и последовательность изложения материала (решения);
- наличие обобщений и выводов, сделанных на основе изучения информационных источников по данной теме;
- предоставление в полном объеме решений имеющихся в задании;
- использование современных способов поиска, обработки и анализа информации;
- самостоятельность выполнения

КРИТЕРИИ ОЦЕНКИ КОНТРОЛЬНОЙ РАБОТЫ

Критериями оценки контрольной работы служат следующие параметры:

– полнота раскрытия темы;

- логика изложения, глубина проработки теоретических основ проблемы и взаимосвязь отдельных рассматриваемых вопросов в контрольной работе;
- качество оформления контрольной работы в соответствии с предъявляемыми требованиями;
- отношение студента к работе.

Оценка «зачтено» выставляется студенту, являющемуся автором контрольной работы, соответствующей всем предъявляемым требованиям, в том числе формальным. При этом в работе студент должен:

а) продемонстрировать умение раскрыть актуальность заявленной темы; проиллюстрировать ее сформулированными теоретическими предложениями;

б) соблюдать логику и последовательность изложения, рассматриваемых вопросов;

в) показать умение анализировать и делать выводы по всему представленному материалу;

г) грамотно и корректно подходить к текстовому материалу.

Оценка «не зачтено» выставляется студенту, являющемуся автором контрольной работы, не соответствующей предъявляемым требованиям. Оценка «не зачтено» выставляется также, если студент:

а) не раскрыл актуальность темы исследования;

б) не предложил теоретических разработок.

Оценка «не зачтено» также выставляется, если возникли обоснованные сомнения в том, что студент не является автором представленной контрольной работы (не ориентируется в тексте работы; не может дать ответы на уточняющие вопросы, касающиеся сформулированных в работе теоретических и практических предложений и т.д.). Такое решение принимается и в том случае, если работа не соответствует предъявляемым требованиям.

СТРУКТУРА КОНТРОЛЬНОЙ РАБОТЫ

Контрольная работа может содержать следующие разделы:
- титульный лист (оформляется в соответствии с Приложением 1);
- основная часть.

В основной части должны быть приведены формулировки каждого задания, подробное описание решения задачи с таблицами, схемами при необходимости. Задания должны выполняться в приведенной последовательности.

Каждое задание должно содержать следующие этапы:
1. Полная формулировка условия.
2. Программный код полного решения задачи.

3. Скриншот программного кода в среде IDEL.
4. Скриншот выполнения программного кода в среде IDEL.
5. Контрольная работа сдается в печатном виде.
5. Электронная версия решения заданий так же должно быть сохранена в файле с расширением ФамилияИмя студента_к_р.ру и сдана на проверку.

ВЫБОР ВАРИАНТА КОНТРОЛЬНОЙ РАБОТЫ

Вариант контрольной работы определяется по последней цифре номера зачетки: 1 - 1 вар; 2 - 2 вар; …9 - 9 вар; 0 - 10 вар.

ВАРИАНТЫ КОНТРОЛЬНОЙ РАБОТЫ

Вариант 1

Задание 1.
Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
"   это    первое предложение. оно    содержит лишние пробелы. последнее предложение имеет ошибки      в регистре  ."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными и убрать лишние пробелы.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Это первое предложение. Оно содержит лишние пробелы. Последнее предложение имеет ошибки в регистре."
```

Задание 2.
В программе задайте переменную budget = 10000 и список цен продуктов prices = [500, 1200, 800, 2000, 1500]. Если цена не превосходит budget, вычесть её из budget и вывести "Куплено", иначе "Пропустить". В конце выведите оставшийся budget.

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со

словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Города и их население.
Ключ: Название города (строка, например, 'Москва').
Значение: Население в миллионах (вещественное число, например, 12.5).
Выведите полные данные по алфавиту для городов в столбец.
Выведите полные данные по возрастанию населения в столбец.

Образец вывода:
```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.
Для каждого задания вам нужно создать множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 5 элементов.
Тема: Цвета радуги.
Элементы: Названия цветов (строки, например, 'красный', 'оранжевый').
Задание: Создайте множество с 7 цветами радуги. Добавьте один дополнительный цвет (например, 'белый'). Создайте второе множество с 3 цветами из первого (тёплые цвета: 'красный', 'оранжевый', 'жёлтый'). Найдите разность (элементы первого, отсутствующие во втором) и проверьте, является ли второе подмножеством первого. Выведите все результаты.
Пример вывода:
Разность: {'белый', 'зелёный', 'голубой', 'синий', 'фиолетовый'}
Подмножество: True

Задание 5.
Напишите код, который вычисляет факториалы чисел 4, 5 и 6, затем вычисляет их сумму и выводит все значения в формате "Факториал 4: 24, Факториал 5: 120, Факториал 6: 720, Сумма: 864".
Пример вывода:
Факториал 4: 24, Факториал 5: 120, Факториал 6: 720, Сумма: 864

Вариант 2

Задание 1.
Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
"первый текст имеет опечатка. второй текст с лишними пробелами. ТРЕТИЙ ТЕКСТ В ВЕРХНЕМ РЕГИСТРЕ."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными или наоборот маленькими, убрать лишние пробелы, исправить опечатку.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Первый текст имеет опечатки. Второй текст с лишними пробелами. Третий текст в верхнем регистре."
```

Задание 2.

Задайте переменную savings = 20000 и список расходов expenses = [800, 1500, 3000, 600, 2500]. Если расход меньше 1000, то "Малый расход" (еда), если 1000-5000 — "Средний" (транспорт), иначе "Большой" (развлечения). Подсчитайте и выведите общее количество расходов по каждой категории и финальный баланс savings (вычитая все расходы).

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Студенты и их оценки.
Ключ: Имя студента (строка, например, 'Алексей').
Значение: Оценка (целое число, например, 85).
Выведите полные данные по алфавиту по именам в столбец.
Выведите полные данные по рейтингу (убыванию) баллов в столбец.
Образец вывода:

```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.
Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Фрукты.
Элементы: Названия фруктов (строки, например, 'яблоко', 'банан').
Задание: Создайте множество с 4 фруктами. Добавьте два новых фрукта. Создайте второе

множество с 3 фруктами (два из первого и один новый). Проверьте наличие одного из добавленных в первом множестве, найдите пересечение с вторым и симметричную разность (элементы, уникальные для каждого множества). Выведите обновлённое первое множество, наличие, пересечение и симметричную разность.

Пример вывода:

Множество: {'яблоко', 'банан', 'апельсин', 'груша', 'виноград', 'киви'}
Наличие груши: True
Пересечение: {'яблоко', 'банан'}
Симметричная разность: {'апельсин', 'груша', 'виноград', 'киви', 'манго'}

Задание 5.

Напишите код, который решает уравнение ln(x) = 1, вычисляя x как экспоненту от 1, и выводит результат в формате "Решение: 2.718282".

Пример вывода:

Решение: 2.718282

Вариант 3

Задание 1.

Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
" это      начало. середина с пробелами .конец без пробела."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными и убрать лишние пробелы, добавить пробелы, после точки.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Это начало. Середина с пробелами. Конец без пробела."
```

Задание 2.

Задайте переменную total_budget = 8000 и список цен одежды items = [1500, 2500, 1800, 3000]. Если цена больше 2000, примените скидку 20% , иначе оставьте как есть. Суммируйте все цены (с учётом скидок) и в конце выведите, превышает ли итоговая сумма total_budget ("Хватит" или "Не хватит").

Задание 3.

В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения

уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.

Тема: Книги и их авторы.
Ключ: Название книги (строка, например, 'Монстры').
Значение: Имя автора (строка, например, 'Серов Максим').
Выведите полные данные по алфавиту по автору в столбец.
Выведите полные данные по алфавиту по названиям в столбец.

Образец вывода:
```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.

Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Животные.
Элементы: Названия животных (строки, например, 'кошка', 'собака').
Задание: Создайте множество с 5 животными. Удалите один элемент (если он существует) и подсчитайте размер множества до и после удаления. Выведите результаты.
Пример вывода:
До удаления: 5
После удаления: 4
Множество: {'кошка', 'собака', 'кролик', 'хомяк', 'рыба'}

Задание 5.

Напишите код, который вычисляет факториал числа 5, затем квадратный корень из него и выводит оба значения в формате "Факториал: 120.0, Корень: 10.954451".
Пример вывода:
Факториал: 120.0, Корень: 10.954451

Вариант 4

Задание 1.

Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
"python - это язык. он используется для программирования. PYTHON ВЕРХНИЙ РЕГИСТР."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными или наоборот маленькими, убрать лишние пробелы или добавить нужные.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Python - это язык. Он используется для программирования. Python верхний регистр."
```

Задание 2.
Задайте переменную vacation_savings = 40000 и список месячных расходов monthly_expenses = [5000, 7000, 3000]. Если расходы свыше 30% от vacation_savings, выведите "Экономьте!", иначе "Всё в порядке". Вычитайте расходы из vacation_savings в цикле и в конце сообщите, хватит ли средств на отпуск (цель — 50000).

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Товары и их цены.
Ключ: Название товара (строка, например, 'Холодильник').
Значение: Цена в рублях (целое число, например, 50 000).
Добавьте два новых товара. Выберите самый дорогой продукт и выведите текст:
Самый дорогой продукт холодильник за 50 000 рублей.
Выведите полные данные по возрастанию цены в столбец.

Образец вывода:

```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.
Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Города.
Элементы: Названия городов (строки, например, 'Москва', 'Санкт-Петербург').
Задание: Создайте множество с 4 городами. Удалите произвольный элемент и выведите его. Затем создайте второе множество с 3 другими городами и объедините оба множества.
Пример вывода:

Удалённый город: Москва
Объединённое множество: {'Санкт-Петербург', 'Екатеринбург', 'Казань', 'Новосибирск', 'Краснодар', 'Владивосток'}

Задание 5.
Напишите код, который вычисляет тангенс угла π/6 радиан, затем арктангенс этого значения и выводит их в формате "Тангенс: 0.577350, Арктангенс: 0.523599".
Пример вывода:
Тангенс: 0.577350, Арктангенс: 0.523599

Вариант 5

Задание 1.
Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
" начало    с пробелами. середина          с множеством пробелов.КОНЕЦ  "
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными и убрать лишние пробелы, добавить точку в конце, если её нет.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Начало с пробелами. Середина с множеством пробелов. Конец."
```

Задание 2.
Задайте список покупок purchases = [(300, 1), (600, 2), (400, 1), (900, 2), (200, 1), (700, 2)], где первый элемент — цена, второй — категория (1 — еда, 2 — быт). Если категория 1 и цена меньше 500, то "Бюджетная еда", иначе "Дорогая". Суммируйте расходы отдельно для каждой категории и в конце выведите, какая категория дороже.

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Страны и их столицы.

Ключ: Название страны (строка, например, 'Russia').
Значение: Название столицы (строка, например, 'Moscow').
Используйте метод словаря выведите столицы существующей страны и для несуществующей страны (с дефолтным значением 'Не найдено').
Выведите оба результата.
Пример вывода:
Столица России: Moscow
Столица несуществующей страны: Не найдено

Задание 4.

Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Дни недели.
Элементы: Названия дней (строки, например, 'понедельник', 'вторник').
Задание: Создайте множество с 7 днями недели. Выведите его, затем очистите и проверьте, равно ли оно пустому множеству. Выведите результат проверки.
Пример вывода:
Множество до очистки: {'понедельник', 'вторник', 'среда', 'четверг', 'пятница', 'суббота', 'воскресенье'}
Равно пустому: True

Задание 5.

Напишите код, который вычисляет логарифм числа 8 по основанию 2, затем возводит 2 в эту степень и выводит оба значения в формате "Логарифм: 3.0, Обратная: 8.0".
Пример вывода:
Логарифм: 3.0, Обратная: 8.0

Вариант 6

Задание 1.

Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
"это    тест. он имеет    ошибки. ВСЕ В НИЖНЕМ."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными или маленькими, убрать лишние пробелы.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Это тест. Он имеет ошибки. Все в нижнем."
```

Задание 2.
Задайте переменную entertainment_budget = 5000 и список стоимостей билетов tickets = [800, 1200, 600, 1500, 900]. Если стоимость не превышает 1000, разрешите покупку и вычтите из entertainment_budget ("Приобретено"), иначе "Слишком дорого, пропустить". В конце выведите количество приобретённых развлечений и остаток entertainment_budget.

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Фильмы и их режиссёры.
Ключ: Название фильма (строка, например, 'Inception').
Значение: Имя режиссёра (строка, например, 'Christopher Nolan').
Задание: Создайте словарь с 5 фильмами и режиссёрами.
Удалите один фильм с помощью и выведите удалённое значение (режиссёра), а затем весь словарь после удаления по алфавиту в столбик.
Образец вывода:
```
Удалённый режиссёр: Christopher Nolan

Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.
Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Спортивные игры.
Элементы: Названия игр (строки, например, 'футбол', 'баскетбол').
Задание: Создайте два множества: первое с 3 играми, второе с 3 другими (с пересечением из 2). Вычислите объединение и пересечение, выведите оба результата.
Пример вывода:
Объединение: {'футбол', 'баскетбол', 'волейбол', 'теннис', 'плавание', 'бег'}
Пересечение: {'футбол', 'баскетбол'}

Задание 5.
Напишите код, который пытается решить квадратное уравнение $x^2 + 2x + 5 = 0$, вычисляя дискриминант, и выводит сообщение "Дискриминант отрицательный: -16.0, корней нет".
Пример вывода:
Дискриминант отрицательный: -16.0, корней нет

Вариант 7

Задание 1.
Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
" первый. второй      с пробелами.третий без пробела."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными и убрать лишние пробелы, добавить пробел перед точкой.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Первый. Второй с пробелами. Третий без пробела."
```

Задание 2.
Задайте список расходов family_expenses = [5000, 12000, 8000, 15000, 6000]. Если расход больше 10000, добавьте "налог" 10% (увеличьте сумму), иначе оставьте без изменений. Суммируйте все расходы и в конце проверьте: если итог превысил 50000, выведите "Превышен лимит семьи", иначе "В пределах нормы".

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Животные и их виды.
Ключ: Название животного (строка, например, 'Lion').
Значение: Вид (строка, например, 'Mammal').
Проверьте, существует ли ключ 'Lion' и выведите True или False.
Также проверьте несуществующий ключ, например, 'Dragon'.

Пример вывода:
Наличие Lion: True
Наличие Dragon: False

Задание 4.
Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.

Тема: Предметы в классе.
Элементы: Названия предметов (строки, например, 'ручка', 'карандаш').
Задание: Создайте два множества: первое с 4 предметами в классе, второе с 4 предметами у учеников (с пересечением из 3). Найдите пересечение и разность (элементы первого, отсутствующие во втором).
Пример вывода:
Пересечение: {'ручка', 'карандаш', 'тетрадь'}
Разность: {'книга'}

Задание 5.

Напишите код, который вычисляет квадратный корень из 25, затем возводит результат в квадрат и выводит оба значения в формате "Корень: 5.0, Квадрат: 25.0".
Пример вывода:
Корень: 5.0, Квадрат: 25.0

Вариант 8

Задание 1.

Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
"код на    python. он простой. PYTHON     ПРОСТОЙ."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными или маленькими, убрать лишние пробелы.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Код на Python. Он простой. Python простой."
```

Задание 2.

Задайте переменную current_savings = 15000 и список цен гаджетов gadgets = [4000, 8000, 6000]. Если цена не превышает половину current_savings, то "Можно купить сейчас" (вычесть цену), иначе "Отложить". В цикле отслеживайте current_savings и в конце выведите, сколько гаджетов куплено и финальную сумму.

Задание 3.

В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения

уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Предметы и их цвета.
Ключ: Название предмета (строка, например, 'Apple').
Значение: Цвет (строка, например, 'Red').
Создайте два словаря: основной с 3 предметами и их цветами, и второй с 2 новыми предметами. Добавьте второй к первому. Выведите обновлённый основной словарь по алфавиту в столбик.
Образец вывода:

```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.

Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Месяцы года.
Элементы: Названия месяцев (строки, например, 'январь', 'февраль').
Задание: Создайте два множества: первое с 12 месяцами, второе с 6 месяцами (с пересечением). Выведите размер первого, проверьте наличие 'июнь', найдите симметричную разность (элементы, присутствующие только в одном из множеств).
Пример вывода:
Размер первого: 12
Наличие июня: True
Симметричная разность: {'август', 'сентябрь', 'октябрь', 'ноябрь', 'декабрь'}

Задание 5.

Напишите код, который вычисляет синус и косинус угла π/4 радиан и выводит их в формате "Синус: 0.707107, Косинус: 0.707107".
Пример вывода:
Синус: 0.707107, Косинус: 0.707107

Вариант 9

Задание 1.

Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
" это    начало. середина с ошибками.КОНЕЦ БЕЗ ТОЧКИ"
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными и убрать лишние пробелы, поставить точку в конце.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.

Пример ожидаемого вывода (после исправлений):

```
"Это начало. Середина с ошибками. Конец без точки."
```

Задание 2.
Задайте переменную weekly_budget = 3500 и список ежедневных трат daily_transport = [400, 600, 800, 300, 700, 500, 900]. Если трата выше 500, то "Перерасход дня" (не вычитайте, предупредите), иначе вычтите из weekly_budget. В конце суммируйте все вычтенные траты и сообщите, был ли бюджет соблюдён (на основе остатка).

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Музыкальные инструменты и их типы.
Ключ: Название инструмента (строка, например, 'Guitar').
Значение: Тип (строка, например, 'String').
Выведите количество элементов в словаре с помощью len(). Также выведите все данные по алфавиту.
Образец вывода:

```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.
Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.
Тема: Овощи.
Элементы: Названия овощей (строки, например, 'морковь', 'картофель').
Задание: Создайте два множества: первое с 5 овощами, второе с 3 овощами (все из первого). Найдите разность (элементы первого, отсутствующие во втором) и проверьте, является ли второе подмножеством первого.
Пример вывода:
Разность: {'морковь', 'лук', 'перец', 'брокколи'}
Подмножество: True

Задание 5.
Напишите код, который вычисляет натуральный логарифм числа 10, затем возводит его в экспоненту и выводит оба значения в формате "Логарифм: 2.302585, Экспонента: 10.0".
Пример вывода:
Логарифм: 2.302585, Экспонента: 10.0

Вариант 10

Задание 1.
Написать программу, которая принимает строку, содержащую три предложения, и исправляет их согласно правилам. Программа должна выводить исправленный текст.
Текст (строка), который нужно обработать, запишите его в переменную.

```
"python      лучший. он для новичков. PYTHON      ДЛЯ НОВИЧКОВ."
```

Нужно исправить (используя методы строк): сделать нужные буквы заглавными или наоборот маленькими, убрать лишние пробелы.
Добавьте комментарии, объясняющие, какие методы вы использовали и почему.
Пример ожидаемого вывода (после исправлений):

```
"Python лучший. Он для новичков. Python для новичков."
```

Задание 2.

Задайте переменную monthly_savings = 12000 и список недельных расходов weekly_expenses = [2500, 1800, 2200, 1600]. Если расходы < 2000, добавьте бонус 10% от сэкономленной суммы, иначе ничего. Вычитайте расходы и накапливайте бонусы. В конце выведите итоговые monthly_savings и "Хорошая экономия" (если бонус выше 0) или "Нужно экономить больше".

Задание 3.
В программе создайте словарь в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Выполните операции со словарём и выведите результаты с помощью print(). Тема, ключи и значения уже указаны в задании. Ваши словари должны иметь минимум 5 пар ключ-значение.
Тема: Спортсмены и их виды спорта.
Ключ: Имя спортсмена (строка, например, 'Usain Bolt').
Значение: Вид спорта (строка, например, 'Athletics'). Выведите словарь по алфавиту спортсменов. Затем выведите данные по видам спорта.
Образец вывода:

```
Ключ1  - значение1
Ключ2 - значение2
```

Задание 4.
Для каждого задания вам нужно создать статические множества в коде на основе указанной темы (не используйте ввод данных от пользователя — input). Каждое задание требует выполнения

операций с множеством и вывода результатов с помощью print(). Тема и элементы множеств уже указаны в задании. Ваши множества должны иметь минимум 3–5 элементов (в зависимости от задания). После решения протестируйте код и убедитесь, что вывод соответствует требованиям.

Тема: Цифры.

Элементы: Названия цифр (строки, например, 'один', 'два').

Задание: Создайте три множества: первое с 5 цифрами, второе с 3 цифрами (все из первого), третье с 2 новыми. Проверьте, является ли второе подмножеством первого, затем объедините первое и третье.

Пример вывода:

Подмножество: True

Объединение с третьим: {'один', 'два', 'три', 'четыре', 'пять', 'шесть', 'семь'}

Задание 5.

Напишите код, который решает квадратное уравнение $x^2 - 5x + 6 = 0$, вычисляя корни, и выводит их в формате "Корни: x1=2.0, x2=3.0".

Пример вывода:

Корни: x1=2.0, x2=3.0

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
УК-4	Способность использовать прикладное программное обеспечение при решении профессиональных задач	1. Использует основные методы и средства получения, представления, хранения и обработки данных.	Знать: □ методы и средства получения, представления, хранения и обработки данных. Уметь: □ получать, представлять, хранить и обрабатывать данные.
		2.Демонстрирует владение профессиональными пакетами прикладных программ.	Знать: □ основные профессиональные пакеты прикладных программ. Уметь: □ использовать профессиональные пакеты прикладных программ.
		3. Выбирает необходимое прикладное программное обеспечение в зависимости от решаемой задачи**.**	Знать: □ рынок прикладного ПО. Уметь: □ формировать критерии

			выборы и осуществлять выбор ПО**.**
		4. Использует прикладное программное обеспечение для решения конкретных прикладных задач**.**	Знать: □ прикладное ПО. Уметь: решать конкретные прикладные задачи.
УК-10	Способность осуществлять поиск, критически анализировать, обобщать и систематизировать информацию, использовать системный подход для решения поставленных задач	1. Четко описывает состав и структуру требуемых данных и информации, грамотно реализует процессы их сбора, обработки и интерпретации	Знать: □ процессы сбора, обработки и интерпретации информации Уметь: □ описывать описывает состав и структуру требуемых данных и информации
		2. Обосновывает сущность происходящего, выявляет закономерности, понимает природу вариабельности	Знать: □ принципы описания закономерностей и вариабельности Уметь: □ обосновывать сущность происходящего
		3. Формулирует признак классификации, выделяет соответствующие ему группы однородных «объектов», идентифицирует общие свойства элементов этих групп, оценивает полноту результатов классификации, показывает прикладное назначение классификационных групп.	Знать: □ подходы к выполнению классификации Уметь: □ выполнять классификацию и оценивать полноту выполнения классификации.
		4. Грамотно, логично, аргументировано формирует собственные суждения и оценки. Отличает факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности	Знать: □ методы аргументации и оценкам. Уметь: □ формировать суждения и оценки.
		5.Аргументи-рованно и логично представляет свою точку зрения посредством и на основе системного описания.	Знать: □ подходы к описанию собственного мнения. Уметь: □ представлять свою точку зрения.

ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ

1. Задания должны выполняться в приведенной последовательности. Каждое задание должно содержать следующие этапы:

1. Полная формулировка условия.

2. Программный код полного решения задачи.

3. Скриншот программного кода в среде IDEL. (Код не должн быть на черном фоне)

4. Скриншот выполнения программного кода в среде IDEL. (Код не должн быть на черном фоне)

2. Контрольная работа сдается в печатном виде.

3. Титульный лист приведен на последней странице.

4. Работа должна быть представлена в печатном виде на листах формата А4 (односторонняя печать). Работа оформляется в текстовом редакторе Word

5. Оформление текста работы: шрифт 14, TimeNewRomans, Код выделен шрифтом Couriers New межстрочное расстояние – 1,5. Страницы, на которых излагается текст, должны иметь поля: верхнее и нижнее – не менее 20 мм; левое – не менее 30 мм; правое – не менее 10 мм; колонтитулы: верхний – 2; нижний – 1,25.

6. Страницы работы следует пронумеровать.

7. Задачи должны выполняться в приведенной последовательности.

8. Контрольная работа сдаётся в папке.

9. Электронная версия решения заданий так же должно быть сохранена в файле с названием ФамилияИмя студента-кр.ру и сдана на проверку: отправлена на электронную почту: gorchakovalla@rambler.ru.

10.. Каждое задание в коде должно быть пронумеровано.

11.Тройные кавычки, закрывающие код, должны быть убраны.

Пример оформления задания.

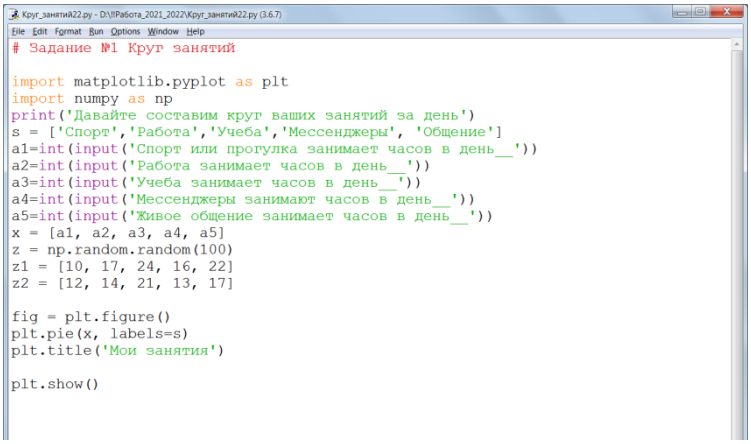
Задача 1.

1. Постройте круговую диаграмму «Круг занятий». В диаграмму должны входить следующие занятия: «Спорт или прогулка»; «Работа»; «Учеба», «Мессенджеры», «Живое общение». Пользователю должно быть предложено указать: сколько часов он тратит на каждое занятие в день. Далее должна быть выведена цветная диаграмма в которой пропорционально отражены занятия.

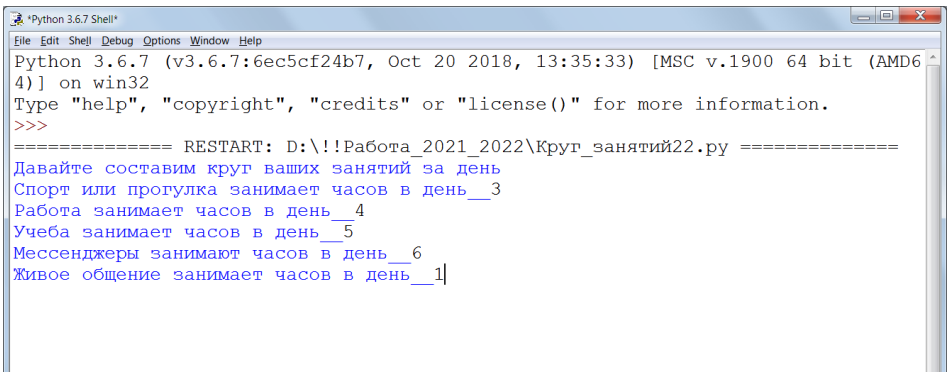
Решение.

Код

```
# Задание №1 Диаграмма "Круг занятий"
import matplotlib.pyplot as plt
import numpy as np
print('Давайте составим круг ваших занятий за день')
s = ['Спорт','Работа','Учеба','Мессенджеры', 'Общение']
a1=int(input('Спорт или прогулка занимает часов в день__'))
a2=int(input('Работа занимает часов в день__'))
a3=int(input('Учеба занимает часов в день__'))
a4=int(input('Мессенджеры занимают часов в день__'))
a5=int(input('Живое общение занимает часов в день__'))
x = [a1, a2, a3, a4, a5]
z = np.random.random(100)
z1 = [10, 17, 24, 16, 22]
z2 = [12, 14, 21, 13, 17]
fig = plt.figure()
plt.pie(x, labels=s)
plt.title('Мои занятия')
plt.show()
```



Выполнение кода


Python 3.6.7 Shell
File Edit Shell Debug Options Window Help
Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20 2018, 13:35:33) [MSC v.1900 64 bit (AMD6
4)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: D:\!!Работа_2021_2022\Круг_занятий22.py ==========
Давайте составим круг ваших занятий за день
Спорт или прогулка занимает часов в день__3
Работа занимает часов в день__4
Учеба занимает часов в день__5
Мессенджеры занимают часов в день__6
Живое общение занимает часов в день__1

Приложение №1

Образец титульного листа контрольной работы

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финуниверситет)
Калужский филиал Финуниверситета
Кафедра «бизнес-информатики и высшей математики»

КОНТРОЛЬНАЯ РАБОТА
по дисциплине «Алгоритмы и структуры данных в языке Python»
на тему: « »
наименование темы или вариант задания

Выполнил (а) студент (ка) _____ курса,
группы _____,
формы обучения_____
(очной, очно- заочной, заочной)
_____
(Ф.И.О. студента)

Проверил преподаватель:
к.п.н., доцент, доцент Костенко Алла Валентиновна

Дата поступления работы на кафедру:

____ _____20 г.

Оценка:

_____ _____
(зачтено/не зачтено) подпись преподавателя

___ _____ 20 г.

Калуга 2025 г