МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОСТОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ РОСТОВСКОЙ ОБЛАСТИ
«РОСТОВСКИЙ-НА-ДОНУ КОЛЛЕДЖ СВЯЗИ И ИНФОРМАТИКИ»

Фонд оценочных средств

по учебной дисциплине

ОП.10 «Основы алгоритмизации и программирования на Python»

программы подготовки специалистов среднего звена

по специальности

09.02.13 «Интеграция решений с применением технологий искусственного интеллекта»

(базовой подготовки)

г. Ростов-на-Дону, 2026 г.

РАССМОТРЕНО
на заседании цикловой комиссии
Программирование
Протокол № 1 от 31.08.2026 года
Председатель ЦК Сулавко С.Н.

УТВЕРЖДАЮ:
Зам. директора по УМР
И.В. Подцатова
«16» ноября 2026 г.

Фонд оценочных средств разработан на основе рабочей программы дисциплины ОП.10 «Основы алгоритмизации и программирования на Python», разработанной по специальности 09.02.13 «Интеграция решений с применением технологий искусственного интеллекта».

Разработчик: Арутюнян М.М. – преподаватель государственного бюджетного профессионального образовательного учреждения Ростовской области «Ростовский-на-Дону колледж связи и информатики»

СОДЕРЖАНИЕ

1. Паспорт фонда оценочных средств 4
2. Результаты освоения дисциплины, подлежащие проверке 5
3. Оценка освоения учебной дисциплины 9
4. Контрольно-оценочные материалы для промежуточной аттестации по учебной дисциплине. 119

1. Паспорт фонда оценочных средств

В результате освоения учебной дисциплины ОП.10 «Основы алгоритмизации и программирования на Python» обучающийся должен обладать предусмотренными ФГОС СПО по специальности 09.02.13 «Интеграция решений с применением технологий искусственного интеллекта» следующими умениями, знаниями и владеть навыками:

Код ОК, ПК	**Уметь**	**Знать**	**Владеть навыками**
ОК.01	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	Методы и подходы решения задач профессиональной деятельности	-
ОК.02	Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии	Основы информационных технологий, методы анализа и интерпретации данных	-
ОК.03	Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности	Основы предпринимательства, правовой и финансовой грамотности, подходы к личностному развитию	-
ОК.04	Эффективно взаимодействовать и работать в коллективе и команде	Основы командной работы, принципы эффективного взаимодействия	-
ОК.05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации	Особенности государственного языка Российской Федерации, правила деловой коммуникации	-
ОК.06	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение	Основы духовно-нравственных ценностей, принципы антикоррупционного поведения	-
ОК.07	Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях	Основы экологии, принципы бережливого производства, методы действий в ЧС	-
ОК.08	Использовать средства физической культуры для поддержания здоровья	Основы физической культуры и здоровья, методы поддержания физической формы	-
ОК.09	Пользоваться профессиональной документацией на	Основы ведения профессиональной документации на разных	-

	государственном и иностранном языках	языках	
ПК.1.8	- использовать методы и приемы формализации поставленных задач - использовать методы и приемы алгоритмизации поставленных задач - использовать программное обеспечение для графического отображения алгоритмов - применять алгоритмы решения типовых задач в соответствующих областях - осуществлять коммуникации с заинтересованными сторонами	- методы и приемы формализации поставленных задач - языки формализации функциональных спецификаций - методы и приемы алгоритмизации поставленных задач - нотации и программное обеспечение для графического отображения алгоритмов - алгоритмы решения типичных задач, области и способы их применения	- составления формализованных описаний решений поставленных задач в соответствии с требованиями технического задания или внутренних документов организации - разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или внутренних документов организации - проверки корректности алгоритмов решения поставленных задач - оценки и согласования сроков выполнения поставленных задач
ПК 3.10	- использовать выбранную среду программирования и средства системы управления базами данных использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода	- синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования - методологии разработки компьютерного программного обеспечения - технологии программирования - компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними	- создавать программный код в соответствии с техническим заданием (готовыми спецификациями) оптимизировать программный код с использованием специализированных программных средств

Формой аттестации в соответствии с рабочей программой по учебной дисциплине ОП.10 «Основы алгоритмизации и программирования на Python» является дифференцированный зачет.

2. Результаты освоения учебной дисциплины, подлежащие проверке

2.1. В результате аттестации по учебной дисциплине ОП.10 «Основы алгоритмизации и программирования на Python» осуществляется комплексная

проверка следующих умений и знаний:

Таблица 1

Результаты обучения	Показатели освоенности компетенций	Методы оценки
Знает: − Методы и подходы решения задач профессиональной деятельности − Основы информационных технологий, методы анализа и интерпретации данных − Основы предпринимательства, правовой и финансовой грамотности, подходы к личностному развитию − Основы командной работы, принципы эффективного взаимодействия − Особенности государственного языка Российской Федерации, правила деловой коммуникации − Основы духовно-нравственных ценностей, принципы антикоррупционного поведения − Основы экологии, принципы бережливого производства, методы действий в ЧС − Основы физической культуры и здоровья, методы поддержания физической формы − Основы ведения профессиональной документации на разных языках − методы и приемы формализации поставленных задач − языки формализации	***Демонстрирует знания:*** − методов и подходов решения задач профессиональной деятельности; − основ информационных технологий, методов анализа и интерпретации данных; − основ предпринимательства, правовой и финансовой грамотности, подходов к личностному развитию; − основ командной работы, принципов эффективного взаимодействия; − особенностей государственного языка Российской Федерации, правил деловой коммуникации; − основ духовно-нравственных ценностей, принципов антикоррупционного поведения; − основ экологии, принципов бережливого производства, методов действий в чрезвычайных ситуациях; − основ физической культуры и здоровья, методов поддержания физической формы; − основ ведения профессиональной документации на разных языках; − методов и приёмов формализации поставленных задач; − языков формализации функциональных спецификаций; − методов и приёмов алгоритмизации поставленных задач; − нотаций и программного обеспечения для графического отображения алгоритмов; − алгоритмов решения типовых задач, областей и способов их применения;	*Экспертное наблюдение выполнения практических работ и видов работ по практике* *Диагностика* *Экзамен*

функциональных спецификаций – методы и приемы алгоритмизации поставленных задач – нотации и программное обеспечение для графического отображения алгоритмов – алгоритмы решения типичных задач, области и способы их применения – синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования – методологии разработки компьютерного программного обеспечения – технологии программирования – компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними Умеет: – Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам – Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии – Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности – Эффективно взаимодействовать и работать в коллективе и команде – Осуществлять устную	– синтаксиса выбранного языка программирования, особенностей программирования на данном языке и его стандартных библиотек; – методологий разработки компьютерного программного обеспечения; – технологий программирования; – компонентов программно-технических архитектур, существующих приложений и интерфейсов взаимодействия с ними. ***Демонстрирует умения:*** – выбирать способы решения задач профессиональной деятельности применительно к различным контекстам; – использовать современные средства поиска, анализа и интерпретации информации, а также информационные технологии; – планировать и реализовывать профессиональное и личностное развитие, применять знания правовой и финансовой грамотности; – эффективно взаимодействовать и работать в коллективе и команде; – осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации; – проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение; – содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях; – использовать средства физической культуры для поддержания здоровья; – пользоваться профессиональной документацией на государственном и иностранном языках; – использовать методы и	

и письменную коммуникацию на государственном языке Российской Федерации − Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение − Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях − Использовать средства физической культуры для поддержания здоровья − Пользоваться профессиональной документацией на государственном и иностранном языках − использовать методы и приемы формализации поставленных задач − использовать методы и приемы алгоритмизации поставленных задач − использовать программное обеспечение для графического отображения алгоритмов − применять алгоритмы решения типовых задач в соответствующих областях − осуществлять коммуникации с заинтересованными сторонами − использовать выбранную среду программирования и средства системы управления базами данных − использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода	приёмы формализации поставленных задач; − применять методы и приёмы алгоритмизации поставленных задач; − использовать программное обеспечение для графического отображения алгоритмов; − применять алгоритмы решения типовых задач в соответствующих областях; − осуществлять коммуникацию с заинтересованными сторонами; − использовать выбранную среду программирования и средства системы управления базами данных; − использовать возможности имеющейся технической и (или) программной архитектуры для написания программного кода.	

3. Оценка освоения учебной дисциплины:

3.1. Формы и методы оценивания

Предметом оценки служат умения и знания, предусмотренные рабочей программой по дисциплине ОП.10 «Основы алгоритмизации и программирования на Python».

Технология оценки знаний и умений по дисциплине соответствует Положению о формах, периодичности и порядке проведения текущего контроля успеваемости и промежуточной аттестации обучающихся в ГБПОУ РО «РКСИ».

Контроль и оценка освоения учебной дисциплины по темам (разделам)

Таблица 2

Элемент учебной дисциплины	**Текущий контроль**		**Промежуточная аттестация**	
	Форма контроля (по темам)	**Проверяемые ОК, ПК**	**Форма контроля (**по разделам**)**	**Проверяемые ОК, ПК**
Раздел 1.Основы программирования на языке Python. Структурное программирование				
Тема1.1 Переменные. Типы объектов. Операции и выражения	Устный опрос, тестирование №1	ОК.01-ОК.09		
Тема1.2 Базовые операторы, конструкции и подпрограммы. Основы структурного программирования	Устный опрос, тестирование №2 «PEP8 и стиль кода», практическое занятие№1-6, контрольная работа №1	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Тема1.3 Файлы	Устный опрос, практическое занятие№7	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Раздел2. Модульное программирование				
Тема2.1Принципы модульного программирования	Устный опрос, практическое занятие №8	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Раздел3. Функциональный стиль программирования в языке Python				
Тема3.1 Итераторы и генераторы функциональный стиль программирования	Устный опрос, практическое занятие №9	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Раздел 4. Библиотеки языка Python				
Тема4.1 Понятие библиотеки, установка и библиотека OS	Устный опрос, практическое занятие №10	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Тема4.2. Библиотека SQlite	Устный опрос, практическое занятие №11	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Раздел5. Основы объектно-ориентированного программирования				

Тема 5.1 Базовые понятия ООП	Устный опрос, практическое занятие №12	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Тема 5.2 Визуальное событийно-управляемое	практическое занятие №13, 14	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
Тема 5.3 Разработка оконного приложения	практическое занятие №15	ОК.01-ОК.09, ПК 1.8, ПК 3.10		
			Дифф.зачет	ОК.01-ОК.09, ПК 1.8, ПК 3.10

3.2. Задания для оценки освоения учебной дисциплины

3.2.1. Задания для оценки знаний и умений в ходе проведения текущего контроля

Темы 1.1

Переменные. Типы объектов. Операции и выражения

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках

Текст задания

Выберите один правильный ответ

1. Понятие алгоритма:

а) Четкая последовательность действий, приводящая к определенному

результату.

b) Программа на любом языке программирования.

c) Процедура или функция для вспомогательных операций.

2. Понятие переменной:

   a) Величина, которая изменяется в процессе работы программы.

   b) Произвольная конструкция языка.

   c) Величина, которая не изменяется в процессе работы программы.

3. Какими скобками изменяется порядок выполнения арифметических операций:

   a) ()

   b) {}

   c) []

4. Какая логическая операция использована: A and B:

   a) Логическое И

   b) Логическое ИЛИ

   c) Логическое отрицание

5. Верное старшинство операций:

   a) <, or, *, +

   b) *, +, <, or

   c) +, <, *, or

6. Что такое Git:

   a) Современный язык программирования

   b) Популярный текстовый редактор

   c) Распределённая система контроля версий

7. Что такое GitHub:

   a) Система управления базами данных

   b) Крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки

   c) Операционная система

8. Этап решения задачи на компьютере, на котором разрабатывается блок-схема:

   a) Постановка задачи

   b) Алгоритмизация

   c) Кодирование

9. Какой тип алгоритма выполняется однократно сверху вниз:

   a) Линейный

   b) Ветвящийся

c) Циклический

10. Что такое интерпретатор:

a) Программа, которая переводит исходный код в машинный и сразу выполняет

b) Программа, которая переводит исходный код в машинный и сохраняет

c) Программа для редактирования текста

Критерии оценки:

- «5» — 9-10 верных ответов
- «4» — 7-8 верных ответов
- «3» — 5-6 верных ответов
- «2» — 4 и менее верных ответов

Тема 1.2
Базовые операторы, конструкции и подпрограммы. Основы структурного программирования

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части 1

Ответьте на вопросы.

Вариант 1

1. **Оператор присваивания:** Чем отличается оператор = в математике от оператора присваивания в программировании? Можно ли написать a = a + 1 и что он делает?
2. **Ввод и вывод данных:** Для чего в Python используются функции input() и print()? Каким типом данных изначально возвращается значение, полученное от input(), и как это изменить?
3. **Условный оператор:** Какую конструкцию используют для проверки трех и более условий (например, «отлично», «хорошо», «плохо»), чтобы не писать вложенные if друг в друга?
4. **Оператор While:** В чем ключевое различие между циклом for и циклом while? Приведите пример ситуации, где while необходим, а for неудобен.
5. **Обработка исключений / Git:** Для чего используется блок try...except? Как назвать команду Git, которая сохраняет изменения локально («фотографирует» текущее состояние проекта)?

Вариант 2

1. **Ввод и вывод данных:** Как вывести данные на одной строке, разделяя их символом ;, а не пробелом? Как при вводе сразу преобразовать строку в целое число?
2. **Условный оператор:** Что такое «вложенный условный оператор»? Чем опасен if x = 5 вместо if x == 5 в языках, допускающих такую запись?
3. **Оператор While и обработка исключений:** Как организовать бесконечный цикл на while и какой оператор используется для его принудительного прерывания? Как с помощью try-except можно корректно выйти из такого цикла при вводе пользователем слова "exit"?
4. **Git и GitHub:** Чем отличается git (программа) от GitHub (веб-сервис)? Назовите

три базовые команды Git для начала работы с репозиторием.

5. **Оператор присваивания:** Что такое множественное присваивание (a, b = 10, 20) и как с его помощью поменять значения двух переменных местами без использования третьей переменной?

Текст задания по теоретической части 2

Ответьте на вопросы.

Вариант 1

1. **Функции:** Что такое функция в программировании и для чего она нужна (назовите минимум две причины)?
2. **Локальные и глобальные переменные:** В чём разница между локальной и глобальной переменной? Где можно обратиться к каждому из этих типов?
3. **Параметры и аргументы:** Чем отличаются параметры функции от аргументов? Что такое «фактические» и «формальные» параметры?
4. **Встроенные функции:** Приведите 3 примера встроенных функций Python (кроме print() и input()) и кратко опишите, что каждая из них делает.
5. **Область видимости:** Что произойдёт, если внутри функции попытаться изменить глобальную переменную без использования ключевого слова global? Будет ли ошибка?

Вариант 2

1. **Локальные переменные:** Может ли локальная переменная иметь то же имя, что и глобальная? Если да, то к какой из них обратится функция внутри себя по умолчанию?
2. **Параметры функции:** Что такое «параметры по умолчанию» (значения по умолчанию) у функции? Приведите пример, когда это удобно.
3. **Возврат значения:** Для чего используется оператор return в функции? Чем отличается функция, которая использует return, от функции, которая его не содержит?
4. **Встроенные функции:** Какая встроенная функция используется для получения длины строки или списка? Какая — для преобразования числа в строку?
5. **Изменение глобальных переменных:** Как внутри функции изменить значение глобальной переменной? Какое ключевое слово для этого нужно использовать в Python?

Текст задания по теоретической части 2

Ответьте на вопросы.

Вариант 1

1. **Создание списков:** Перечислите 3 различных способа создать пустой список в Python. Как создать список из заранее заданных элементов (например, чисел от 1 до 5)?
2. **Методы списков:** Чем отличаются методы append() и extend() для списков? Приведите пример, когда один не может заменить другой.
3. **Цикл for и списки:** Как с помощью цикла for перебрать все элементы списка? Как получить не только значение элемента, но и его индекс (порядковый номер) внутри цикла?
4. **Создание строк:** Какие 3 способа создания строки (какие кавычки можно использовать) существуют в Python? Чем отличается многострочная строка от обычной?
5. **Методы строк:** Как удалить лишние пробелы в начале и в конце строки? Как перевести строку в верхний регистр? Приведите соответствующие методы.

Вариант 2

1. **Индексация списков:** Как обратиться к последнему элементу списка, не зная его длины? Что такое срез (slice) списка и как получить элементы с индекса 2 по 5 (не включая 5)?
2. **Методы списков:** Какие методы позволяют удалить элемент из списка? Назовите минимум 3 разных способа удаления и кратко опишите разницу между ними.
3. **Цикл for и диапазоны:** Как с помощью цикла for и функции range() создать список чисел от 10 до 1 (в обратном порядке)? Как это можно сделать ещё проще, используя срез списка с шагом?
4. **Операции со строками:** Какие операторы можно применять к строкам (например, для объединения или повторения)? Что выведет код: "Hi" * 3 и "Hello" + " " + "World"?
5. **Методы строк:** Как разбить строку на список подстрок по пробелам? Как заменить все вхождения одной подстроки на другую? Как проверить, состоит ли строка только из цифр?

«5» — 5 верных ответов

«4» — 4 верных ответов

«3» — 3 верных ответов

«2» — 0-2 верных ответов

Текст задания по практической части

Практическое занятие №1

Знакомство и работа с интерпретатором Python 3.8. Составление программ линейной структуры

Цели: выработка первичных навыков работы с интерактивной средой, выработка первичных навыков структурного программирования, составление программ линейной структуры.

Порядок работы:

1. В папке Документы создать собственную папку. В нее необходимо размещать все файлы, полученные в результате работы.
2. Через командную строку Windows запустить интерпретатор Python:
   Пуск → Командная строка → Python
3. Увеличить размеры окна по высоте.
4. Использовать Python как калькулятор. Набрать и вычислить арифметические выражения с операциями +, -, *, / и их комбинациями.
5. В зависимости от варианта выполнить задание.

Например, математическую формулу представить соответствующим арифметическим выражением в Python и выполнить расчеты, присвоив переменным начальные значения: **a + bx + cyz (**один символ – одна переменная)

Решение:

1 способ

```
>>>a=1;b=2;x=1;c=2;y=1;z=2
>>>a+b*x+c*y*z
>>>7
```

2 способ

```
>>>a,b,x,c,y,z = 1,2,1,2,1,2
>>>a+b*x+c*y*z
>>>7
```

Варианты заданий

1. 2+ a/b/c
2. 3abc – 8c
3. 102 – df – yz
4. Z-x*4+6a
5. 17 – sfe-45q

6. (25 - rw)/d – 5
7. D – (0.58a + d)/s
8. Af - gr/k
9. 999t – abc/4
10. Q6w + 17ax
11. 17(a+b+d)
12. Dfr/(g+48)
13. Ft - 23yt/8
14. Ur + dcv - 5
15. 23s – 4ac/d
16. 12d – 3(f + 2s)
17. 8lkd – 15
18. (a+b)*c/10*d
19. 0.88ab – 2df
20. 45fr/(ra – 4d)
21. 4*k*(d/6-t)
22. 356d + 14rt
23. 45f – 45w – 45t
24. 87r*19g-4df
25. Rg/15-(d+8t)
26. 62fg/(6t-14s)
27. 0.56gh-34df/45
28. 11.37kl*12.45fd-3
29. 78/5kj/35p-18
30. Deo-878.1gh+100
31. (48we-61as)/20w
32. (18tr+16hj)/41k
33. 45qw+8d-14

Программу и результат перенести в конспект.

6. Выйти из интерпретатора Python
>>>exit()
7. Работа в IDLE. Через командную строку Windows запустить IDLE Python:
Пуск → Командная строка → IDLE
8. Создать новый файл:
File → New File
Сохранить его с именем «Fish». Набрать следующий программный код:

```
print ('><{{{{*>')
print ('          <o((((><')
print
print ('><))))O>')
print ('          <o((((><')
```

9. Сохранить изменения. Выполнить код – F5.
10. Создать новый файл и решить задачу из п.6 используя операторы input и print.
11. Выйти из IDLE Python.

Инструкция к практическому заданию

Пояснения. Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Оформить в электронном виде отчет (.pdf) по практическому занятию. Отчет должен содержать постановку задачи, блок-схему алгоритма с указанием типа алгоритма, текст программы на Python, протокол работы программы.

Пример решения задачи

```
25  a = int(input("Введи переменную a:  "))
26  b = int(input("Введи переменную b:  "))
27  sum = a + b
28  print("Переменная a:", a)
29  print("Переменная b:", b)
30  print(sum)
```

Варианты заданий

Вариант 1. Известно, что X кг конфет стоит A рублей. Определить, сколько стоит 1 кг и Y кг этих же конфет.

Вариант 2. Известно, что X кг шоколадных конфет стоит A рублей, а Y кг ирисок стоит B рублей. Определить, сколько стоит 1 кг шоколадных конфет, 1 кг ирисок, а также во сколько раз шоколадные конфеты дороже ирисок.

Вариант 3. Скорость лодки в стоячей воде V км/ч, скорость течения реки U км/ч (U < V). Время движения лодки по озеру T_1 ч, а по реке (против течения) — T_2 ч. Определить путь S, пройденный лодкой (путь = время • скорость). Учесть, что при движении против течения скорость лодки уменьшается на величину скорости течения.

Вариант 4. Скорость первого автомобиля Vi км/ч, второго — V2 км/ч, расстояние между ними S км. Определить расстояние между ними через T часов, если автомобили удаляются друг от друга. Данное расстояние равно сумме начального расстояния и общего пути, проделанного автомобилями; общий путь = время • суммарная скорость.

Вариант 5. Скорость первого автомобиля Vi км/ч, второго — V2 км/ч, расстояние между ними S км. Определить расстояние между ними через T часов, если автомобили первоначально движутся навстречу друг другу. Данное расстояние равно модулю разности начального расстояния и общего пути, проделанного автомобилями; общий путь = время • суммарная скорость.

Вариант 6. Дано расстояние L в сантиметрах. Используя операцию деления нацело, найти количество полных метров в нем (1 метр =100 см).

Вариант 7. Дана масса M в килограммах. Используя операцию деления нацело, найти количество полных тонн в ней (1 тонна = 1000 кг).

Вариант 8. Дан размер файла в байтах. Используя операцию деления нацело, найти количество полных килобайтов, которые занимает данный файл (1 килобайт = 1024 байта).

Вариант 9. Даны целые положительные числа A и B (A > B). На отрезке длины A размещено максимально возможное количество отрезков длины B (без

наложений). Используя операцию деления нацело, найти количество отрезков B, размещенных на отрезке A.

Вариант 10. Даны целые положительные числа A и B (A > B). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Используя операцию взятия остатка от деления нацело, найти длину незанятой части отрезка A.

Вариант 11. Дано двузначное число. Вывести вначале его левую цифру (десятки), а затем — его правую цифру (единицы). Для нахождения десятков использовать операцию деления нацело, для нахождения единиц — операцию взятия остатка от деления.

Вариант 12. Дано двузначное число. Найти сумму и произведение его цифр.

Вариант 13. Дано двузначное число. Вывести число, полученное при перестановке цифр исходного числа.

Вариант 14. Дано трехзначное число. Используя одну операцию деления нацело, вывести первую цифру данного числа (сотни).

Вариант 15. Дано трехзначное число. Вывести вначале его последнюю цифру (единицы), а затем — его среднюю цифру (десятки).

Вариант 16. Дано трехзначное число. Найти сумму и произведение его цифр.

Вариант 17. Дано трехзначное число. Вывести число, полученное при прочтении исходного числа справа налево.

Вариант 18. Дано трехзначное число. В нем зачеркнули первую слева цифру и приписали ее справа. Вывести

Вариант 19. Дано трехзначное число. В нем зачеркнули первую справа цифру и приписали ее слева. Вывести полученное число.

Вариант 20. Дано трехзначное число. Вывести число, полученное при перестановке цифр сотен и десятков исходного числа (например, 123 перейдет в 213).

Вариант 21. Дано трехзначное число. Вывести число, полученное при перестановке цифр десятков и единиц исходного числа (например, 123 перейдет в 132).

Вариант 22. Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду сотен в записи этого числа.

Вариант 23. Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду тысяч в записи этого числа.

Вариант 24. С начала суток прошло N секунд (N — целое). Найти количество полных минут, прошедших с начала суток.

Вариант 25. С начала суток прошло N секунд (N — целое). Найти количество полных часов, прошедших с начала суток.

Вариант 26. С начала суток прошло N секунд (N — целое). Найти количество секунд, прошедших с начала последней минуты.

Вариант 27. С начала суток прошло N секунд (N — целое). Найти количество секунд, прошедших с начала последнего часа.

Вариант 28. С начала суток прошло N секунд (N — целое). Найти количество полных минут, прошедших с начала последнего часа.

Вариант 29. Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, ..., 6 — суббота. Дано целое число K,

лежащее в диапазоне 1-365. Определить номер дня недели для K-го дня года, если известно, что в этом году 1 января было понедельником.

Вариант 30. Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, . . . , 6 — суббота. Дано целое число K, лежащее в диапазоне 1-365. Определить номер дня недели для K-го дня года, если известно, что в этом году 1 января было четвергом.

Вариант 31. Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, ..., 6 — суббота, 7 — воскресенье. Дано целое число K, лежащее в диапазоне 1-365. Определить номер дня недели для K-го дня года, если известно, что в этом году 1 января было вторником.

Вариант 32. Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, ..., 6 — суббота, 7 — воскресенье. Дано целое число K, лежащее в диапазоне 1-365. Определить номер дня недели для K-го дня года, если известно, что в этом году 1 января было субботой.

Вариант 33. Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, ..., 6 — суббота, 7 — воскресенье. Дано целое число K, лежащее в диапазоне 1-365, и целое число N, лежащее в диапазоне 1-7. Определить номер дня недели для K-го дня года, если известно, что в этом году 1 января было днем недели с номером N.

Вариант 34. Даны целые положительные числа A, B, C. На прямоугольнике размера A x B размещено максимально возможное количество квадратов со стороной C (без наложений). Найти количество квадратов, размещенных на прямоугольнике, а также площадь незанятой части прямоугольника.

Вариант 35. Дан номер некоторого года (целое положительное число). Определить соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.

Критерии оценивания:

Оценка «5» - код программы и отчет предоставлены, задача решена полностью, в соответствии с условием и пояснениями.

Оценка «4» - код программы и отчет предоставлены, задача решена полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Практическое занятие №2

Составление программ ветвящейся структуры на Python

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, **приобрести навыки** составление программ ветвящейся структуры **в** IDE PyCharm Community.

Пример решения задачи

```
# Дано целое число К. Вывести строку-описание оценки, соответствующей числу К
# (1 – «плохо», 2 – «неудовлетворительно», 3 – «удовлетворительно»,
# 4 – «хорошо», 5 – «отлично»). Если К не лежит в диапазоне 1-5, то вывести строку «ошибка».

k = input("Введите число: ")

while type(k) != int:  # обработка исключений
    try:
        k = int(k)
    except ValueError:
        print("Неправильно ввели!")
        k = input("Введите число: ")

if k == 1: print('Плохо')
elif k == 2: print('Неудовлетворительно')
elif k == 3: print('Удовлетворительно')
elif k == 4: print('Хорошо')
elif k == 5: print('Отлично')
else: print('Нет такой оценки!')
```

Инструкция к практическому заданию

Пояснения. Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, блок-схему алгоритма с указанием типа алгоритма, текст программы на Python, протокол работы программы.

Варианты заданий

Вариант 1.

1. Дано целое число А. Проверить истинность высказывания: «Число А является положительным».
2. Даны координаты точки, не лежащей на координатных осях OX и OY. Определить номер координатной четверти, в которой находится данная точка.

Вариант 2.

1. Дано целое число А. Проверить истинность высказывания: «Число А является нечетным».
2. При открытии вклада в банке установлены следующие годовые проценты: при вкладе до 50000р. процент составит 4%; при вкладе от 50000р. до 100000р. процент составит 5%; при вкладе от 100000р. до 150000р. скидка составит 6%; при вкладе от 150000 р. до 200000 р. процент составит 7%. Составить программу, определяющую процентной ставки в зависимости от вносимой суммы.

Вариант 3.

1. Дано целое число А. Проверить истинность высказывания: «Число А является четным».
2. Размер скидки на продукты определен следующим образом: при покупке до 500 р. скидка составит 2%; при покупке от 500 р. до 1000 р. скидка составит 3%; при покупке от 1000 р. до 1500 р. скидка составит 4%; при покупке от 1500 р. до 2000 р. скидка составит 5%. Составить программу определяющую размер скидки в зависимости от потраченной суммы.

Вариант 4.

1. Даны два целых числа: A, B. Проверить истинность высказывания: «Справедливы неравенства A > 2 и B < 3».
2. Спектр видимого излучения представлен в таблице. Составить программу, определяющую название цвета в зависимости от введенной длины волны.

Цвет	Диапазон длин волн, нм
Фиолетовый	≤450
Синий	450—480
Сине-зелёный	480—510
Зелёный	510—550
Жёлто-зелёный	550—570
Жёлтый	570—590
Оранжевый	590—630
Красный	≥630

Вариант 5.

1. Даны два целых числа: A, B. Проверить истинность высказывания: «Справедливы неравенства A > 0 или B < —2».
2. Мастям игральных карт присвоены порядковые номера: 1– пики, 2 – трефы, 3 – бубны, 4 – червы. Достоинству карт, старших десятки, присвоены номера: 11 – валет, 12 – дама, 13 – король, 14 – туз. Дано трехзначное число, в котором первая цифра указывает на масть, а вторые две на достоинство карты. Вывести соответствующее название карты вида «дама червей», «туз треф» и т.п.

Вариант 6.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Справедливо двойное
2. Смоделировать простейший калькулятор, умеющий выполнять 4 основные арифметические операции

Вариант 7.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Число B находится между числами A и C».
2. Единицы массы пронумерованы следующим образом: 1 — килограмм, 2 — миллиграмм, 3 — грамм, 4 — тонна, 5 — центнер. Дан номер единицы массы (целое число в диапазоне 1-5) и масса тела в этих единицах (вещественное число). Найти массу тела в килограммах.

Вариант 8.

1. Даны два целых числа: A, B. Проверить истинность высказывания: «Каждое из чисел A и B нечетное».
2. Единицы длины пронумерованы следующим образом: 1 — дециметр, 2 — километр, 3 — метр, 4 — миллиметр, 5 — сантиметр. Дан номер единицы длины (целое число в диапазоне 1-5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах.

Вариант 9.

1. Даны два целых числа: A, B. Проверить истинность высказывания: «Хотя бы одно из чисел A и B нечетное».
2. Арифметические действия над числами пронумерованы следующим образом: 1 — сложение, 2 — вычитание, 3 — умножение, 4 — деление. Дан номер действия N

(целое число в диапазоне 1-4) и вещественные числа A и B (B не равно 0). Выполнить над числами указанное действие и вывести результат.

Вариант 10.

1. Даны два целых числа: A, B. Проверить истинность высказывания: «Ровно одно из чисел A и B нечетное».
2. Дан номер месяца — целое число в диапазоне 1-12 (1 — январь, 2 — февраль и т. д.). Определить количество дней в этом месяце для невисокосного года.

Вариант 11.

1. Даны два целых числа: A, B. Проверить истинность высказывания: «Числа A и B имеют одинаковую четность».
2. Дано целое число, лежащее в диапазоне 1-999. Вывести его строку- описание вида «четное двузначное число», «нечетное трехзначное число» и т. д.

Вариант 12.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Каждое из чисел A, B, C положительное».
2. Дано целое число. Вывести его строку-описание вида «отрицательное четное число», «нулевое число», «положительное нечетное число» и т. д.

Вариант 13.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Хотя бы одно из чисел A, B, C положительное».
2. Дан номер года (положительное целое число). Определить количество дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).

Вариант 15.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Ровно одно из чисел A, B, C положительное».
2. Даны целочисленные координаты точки на плоскости. Если точка совпадает с началом координат, то вывести 0. Если точка не совпадает с началом координат, но лежит на оси OX или OY, то вывести соответственно 1 или 2. Если точка не лежит на координатных осях, то вывести 3.

Вариант 16.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Ровно два из чисел A, B, C являются положительными».
2. Даны три целых числа, одно из которых отлично от двух других, равных между собой. Определить порядковый номер числа, отличного от остальных.

Вариант 17.

1. Дано целое положительное число. Проверить истинность высказывания: «Данное число является четным двузначным».

2. Даны три переменные вещественного типа: A, B, C. Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных A, B, C.

Вариант 18.

1. Дано целое положительное число. Проверить истинность высказывания: «Данное число является нечетным трехзначным».
2. Даны три числа. Найти среднее из них (то есть число, расположенное между наименьшим и наибольшим).

Вариант 19.

1. Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара совпадающих».
2. Даны две переменные целого типа: A и B. Если их значения не равны, то присвоить каждой переменной сумму этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных A и B.

Вариант 20.

1. Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара взаимно противоположных».
2. Даны два числа. Вывести вначале большее, а затем меньшее из них.

Вариант 21.

1. Дано трехзначное число. Проверить истинность высказывания: «Все цифры данного числа различны».
2. Даны два числа. Вывести порядковый номер меньшего из них.

Вариант 22.

1. Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую последовательность».
2. Даны два числа. Вывести большее из них.

Вариант 23.

1. Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую или убывающую последовательность».
2. Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае вычесть из него 2. Вывести полученное число.

Вариант 24.

1. Дано четырехзначное число. Проверить истинность высказывания: «Данное число читается одинаково слева направо и справа налево».
2. Даны координаты двух различных полей шахматной доски x1, y1, x2, y2 (целые числа, лежащие в диапазоне 1-8). Проверить истинность высказывания: «Ладья за один ход может перейти с одного поля на другое».

Вариант 25.

1. Даны числа x, y. Проверить истинность высказывания: «Точка с координатами (x, y) лежит во второй координатной четверти».
2. Даны координаты поля шахматной доски x, y (целые числа, лежащие в диапазоне 1-8). Учитывая, что левое нижнее поле доски (1,1) является черным, проверить истинность высказывания: «Данное поле является белым».

Вариант 26.

1. Даны числа x, y. Проверить истинность высказывания: «Точка с координатами (x, y) лежит в четвертой координатной четверти».
2. Даны целые числа a, b, c. Проверить истинность высказывания: «Существует треугольник со сторонами a, b, c».

Вариант 27.

1. Даны числа x, y. Проверить истинность высказывания: «Точка с координатами (x, y) лежит во второй или третьей координатной четверти».
2. Дан номер месяца — целое число в диапазоне 1-12 (1 — январь, 2 — февраль и т. д.). Вывести название соответствующего времени года («зима», «весна», «лето», «осень»).

Вариант 28.

1. Даны числа x, y. Проверить истинность высказывания: «Точка с координатами (x, y) лежит в первой или третьей координатной четверти».
2. Дано целое число в диапазоне 1-7. Вывести строку — название дня недели, соответствующее данному числу (1 — «понедельник», 2 — «втор

Вариант 29.

1. Даны числа x, y, x_1, y_1, x_2, y_2. Проверить истинность высказывания: «Точка с координатами (x, y) лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты (x1, y1), правая нижняя — (x2, y2), а стороны параллельны координатным осям».
2. На числовой оси расположены три точки: A, B, C. Определить, какая из двух последних точек (B или C) расположена ближе к A, и вывести эту точку и ее расстояние от точки A.

Вариант 30.

1. Даны целые числа a, b, c, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами a, b, c является равносторонним».
2. Даны четыре целых числа, одно из которых отлично от трех других, равных между собой. Определить порядковый номер числа, отличного от остальных.

Вариант 31.

1. Даны целые числа a, b, c, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами a, b, c является равнобедренным».
2. Даны три числа. Найти сумму двух наибольших из них.

Вариант 32.

1. Даны целые числа a, b, c, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами a, b, c является прямоугольным».
2. Даны три числа. Вывести вначале наименьшее, а затем наибольшее из данных чисел.

Вариант 33.

1. Даны координаты двух различных полей шахматной доски x1, y1, x2, y2 (целые числа, лежащие в диапазоне 1-8). Проверить истинность высказывания: «Данные поля имеют одинаковый цвет».
2. Даны три числа. Найти наименьшее из них.

Вариант 34.

1. Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае не изменять его. Вывести полученное число.
2. Даны две переменные целого типа: A и B. Если их значения не равны, то присвоить каждой переменной большее из этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных A и B.

Вариант 35.

1. Дано целое число. Если оно является положительным, то прибавить к нему 1; если отрицательным, то вычесть из него 2; если нулевым, то заменить его на 10. Вывести полученное число.
2. Даны три целых числа. Найти количество положительных и количество отрицательных чисел в исходном наборе.

Критерии оценивания:

Оценка «5» - решены две задачи, коды программ и отчет предоставлены, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет предоставлены, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Практическое занятие №3

Составление программ циклической структуры на Python. Размещение проекта на GitHub

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыкисоставление программ ветвящейся структуры в IDE PyCharm Community, первичные навыки работы с системой контроля версий GitHub.

Пример решения задачи

```
# Даны два целых числа A и B (A < B). Вывести в порядке возрастания все целые числа,
# расположенные между A и B (включая сами числа A и B), а также количество этих чисел
# (использовать оператор цикла)

a, b = input("Введите первое число: "), input("Введите второе число: ")

while type(a) != int:  # обработка исключений
    try:
        a = int(a)
    except ValueError:
        print("Неправильно ввели!")
        a = input("Введите первое число: ")

while type(b) != int:  # обработка исключений
    try:
        b = int(b)
    except ValueError:
        print("Неправильно ввели!")
        b = input("Введите второе число: ")
k = 0
while a <= b:
    print(a)
    a += 1
    k += 1
print('Количество чисел:  ', k)
```

Пояснения.

Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, блок-схему алгоритма с указанием типа алгоритма, текст программы на Python, протокол работы программы.

Разместить на GitHub проект и отчет по практическому занятию.

Вариант 1.

1. Даны два целых числа A и B (A < B). Вывести в порядке возрастания все целые числа, расположенные между A и B (включая сами числа A и B), а также количество N этих чисел.
2. Даны положительные числа A, B, C. На прямоугольнике размера A x B размещено максимально возможное количество квадратов со стороной C (без наложений). Найти количество квадратов, размещенных на прямоугольнике. Операции умножения и деления не использовать.

Вариант 2.

1. Даны два целых числа A и B (A < B). Вывести в порядке убывания все целые числа, расположенные между A и B (не включая числа A и B), а также количество N этих чисел.
2. Дано целое число N (>0). С помощью операций деления нацело и взятия остатка от деления определить, имеются ли в записи числа N нечетные цифры. Если имеются, то вывести TRUE, если нет — вывести FALSE.

Вариант 3.

1. Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1, 2, ..., 10 кг конфет.

2. Дано целое число N (>0). С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа N цифра «2». Если имеется, то вывести TRUE, если нет — вывести FALSE.

Вариант 4.
1. Дано вещественное число — цена 1 кг конфет. Вывести стоимость 0.1, 0.2, ..., 1 кг конфет.
2. Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти число, полученное при прочтении числа N справа налево.

Вариант 5.
1. Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1.2, 1.4, ..., 2 кг конфет.
2. Дано целое число N (>0). Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.

Вариант 6.
1. Даны два целых числа A и B (A < B). Найти сумму всех целых чисел от A до B включительно.
2. Дано целое число N (>0). Используя операции деления нацело и взятия остатка от деления, вывести все его цифры, начиная с самой правой (разряда единиц).

Вариант 7.
1. Даны два целых числа A и B (A < Б). Найти произведение всех целых чисел от A до B включительно.
2. Спортсмен-лыжник начал тренировки, пробежав в первый день 10 км. Каждый следующий день он увеличивал длину пробега на P процентов от пробега предыдущего дня (P — вещественное, 0< P <50). По данному P определить, после какого дня суммарный пробег лыжника за все дни превысит 200 км, и вывести найденное количество дней K (целое) и суммарный пробег S (вещественное число).

Вариант 8.
1. Даны два целых числа A и B (A < B). Найти сумму квадратов всех целых чисел от A до B включительно.
2. Начальный вклад в банке равен 1000 руб. Через каждый месяц размер вклада увеличивается на P процентов от имеющейся суммы (P — вещественное число, 0< P <25). По данному P определить, через сколько месяцев размер вклада превысит 1100 руб., и вывести найденное количество месяцев K (целое число) и итоговый размер вклада S (вещественное число).

Вариант 9.
1. Дано целое число N (>0). Найти сумму 1 + 1/2 + 1/3 + ... + 1/N
2. Дано число A (> 1). Вывести наименьшее из целых чисел K, для которых сумма 1 + 1/2 + ... + 1/K будет больше A, и саму эту сумму.

Вариант 10.

1. Дано целое число N (> 0). Найти сумму $N^2 + (N + 1)^2 + (N + 2)^2 + ... + (2N)^2$
2. Дано целое число N (> 1). Найти наибольшее целое число K, при котором выполняется неравенство $3^K < N$.

Вариант 11.
1. Дано целое число N (>0). Найти произведение 1.1 • 1.2 • 1.3 •... (N сомножителей).
2. Дано целое число N (> 1). Найти наименьшее целое число K, при котором выполняется неравенство $3^K > N$.

Вариант 12.
1. Дано целое число N (>0). Найти значение выражения 1.1 - 1.2 + 1.3 - ... (N слагаемых, знаки чередуются). Условный оператор не использовать.
2. Дано целое число N (> 1). Вывести наибольшее из целых чисел K, для которых сумма 1 + 2 + ... + K будет меньше или равна N, и саму эту сумму.

Вариант 13.
1. Дано вещественное число A и целое число N (>0). Найти A в степени N: A^N = AA ... •A (числа A перемножаются N раз).
2. Дано целое число N (> 1). Вывести наименьшее из целых чисел K, для которых сумма 1 + 2 + . . . + K будет больше или равна N, и саму эту сумму.

Вариант 14.
1. Дано вещественное число A и целое число N (>0). Используя один цикл, вывести все целые степени числа A от 1 до N.
2. Дано целое число N (>0). Найти наименьшее целое положительное число K, квадрат которого превосходит N: $K^2 > N$. Функцию извлечения квадратного корня не использовать.

Вариант 15.
1. Дано вещественное число A и целое число N (>0). Используя один цикл, найти сумму $1 + A + A^2 + A^3 + ... + A^N$.
2. Дано целое число N (>0), являющееся некоторой степенью числа 2: $N = 2^K$. Найти целое число K — показатель этой степени.

Вариант 16.
1. Дано вещественное число A и целое число N (>0). Используя один цикл, найти значение выражения $1 - A + A^2 - A^3 + ... +(-1)^N A^N$. Условный оператор не использовать.
2. Даны положительные числа A и B (A > Б). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Не используя операции умножения и деления, найти количество отрезков Б, размещенных на отрезке A.

Вариант 17.
1. Дано целое число N (>0). Используя один цикл, найти сумму 1 + 1/(1!) + 1/(2!) + 1/(3!) + ... + 1/(N!) (выражение N! — N-факториал — обозначает произведение всех целых чисел от 1 до N: N! = 1-2-.. . N). Полученное число является приближенным значением константы e = exp(1).

2. Даны положительные числа A и B (A > Б). На отрезке длины A размещено максимально возможное количество отрезков длины Б (без наложений). Не используя операции умножения и деления, найти длину незанятой части отрезка A.

Вариант 18.

1. Дано вещественное число X и целое число N (> 0). Найти значение выражения $1 + X + X^2/(2!) + ... + X^N/(N!)$ (N! = 12 ...N). Полученное число является приближенным значением функции exp в точке X.
2. Даны целые положительные числа A и B (A < B). Вывести все целые числа от A до B включительно; при этом каждое число должно выводиться столько раз, каково его значение (например, число 3 выводится 3 раза).

Вариант 19.

1. Дано вещественное число X и целое число N (> 0). Найти значение выражения $X - X^3/(3!) + X^5/(5!) - ... + (-1)^N \cdot X^{2-N+1}/((2-N+1)!)$ (N! = 12 ...N). Полученное число является приближенным значением функции sin в точке X.
2. Дано целое число N (> 0). Найти сумму $1^N + 2^{N-1} + ... + N^1$.

Вариант 20.

1. Дано вещественное число X и целое число N (> 0). Найти значение выражения $1 - X^2/(2!) + X^4/(4!) - ... + (-1)^N \cdot X^{2*N}/((2-N)!)$ (N! = 12 ...N). Полученное число является приближенным значением функции cos в точке X.
2. Дано целое число N (> 0). Найти сумму $1^1 + 2^2 + ... + N^N$.

Вариант 21.

1. Дано вещественное число X (|X|<1) и целое число N (>0). Найти значение выражения $X - X^2/2 + X^3/3 - ... + (-1)^{N-1}X^N/N$. Полученное число является приближенным значением функции ln в точке 1 + X.
2. Дано целое число N (>0). Если оно является степенью числа 3, то вывести TRUE, если не является — вывести FALSE.

Вариант 22.

1. Дано вещественное число X (|X|<1) и целое число N (>0). Найти значение выражения $X - X^3/3 + X^5/5 - ... + (-1)^N X^{2N+1}/(2N+1)$. Полученное число является приближенным значением функции arctg в точке X.
2. Дано число A (>1). Вывести наибольшее из целых чисел K, для которых сумма 1 + 1/2 + ... + 1/K будет меньше A, и саму эту сумму.

Вариант 23.

1. Даны целые положительные числа N и K. Найти сумму $1^K + 2^K + ... + N^K$.
2. Дано целое число N (>0). Найти наибольшее целое число K, квадрат которого не превосходит N: $K^2 < N$. Функцию извлечения квадратного корня не использовать.

Вариант 24.

1. Даны целые числа A и B (A < B). Вывести все целые числа от A до B включительно; при этом число A должно выводиться 1 раз, число A + 1 должно выводиться 2 раза и т. д.
2. Даны целые положительные числа N и K. Используя только операции сложения и вычитания, найти частное от деления нацело N на K, а также остаток от этого деления.

Вариант 25.
1. Даны два целых числа A и B (A < B). Вывести в порядке возрастания все целые числа, расположенные между A и B (включая сами числа A и B), а также количество N этих чисел.
2. Дано целое число N (>0). Если оно является степенью числа 3, то вывести TRUE, если не является — вывести FALSE.

Вариант 26.
1. Даны два целых числа A и B (A < B). Вывести в порядке убывания все целые числа, расположенные между A и B (не включая числа A и B), а также количество N этих чисел.
2. Дано целое число N (> 1). Вывести наибольшее из целых чисел K, для которых сумма 1 + 2 + ... + K будет меньше или равна N, и саму эту сумму.

Вариант 27.
1. Дано целое число N (>0). Найти сумму 1 + 1/2 + 1/3 + ... + 1/N
2. Дано целое число N (> 1). Вывести наименьшее из целых чисел K, для которых сумма 1 + 2 + . . . + K будет больше или равна N, и саму эту сумму.

Вариант 28.
1. Дано целое число N (> 0). Найти сумму $N^2 + (N + 1)^2 + (N + 2)^2 + ... + (2N)^2$
2. Начальный вклад в банке равен 1000 руб. Через каждый месяц размер вклада увеличивается на P процентов от имеющейся суммы (P — вещественное число, 0< P <25). По данному P определить, через сколько месяцев размер вклада превысит 1100 руб., и вывести найденное количество месяцев K (целое число) и итоговый размер вклада S (вещественное число).

Вариант 29.
1. Дано целое число N (>0). Найти произведение 1.1 • 1.2 • 1.3 •... (N сомножителей).
2. Даны положительные числа A, B, C. На прямоугольнике размера A x B размещено максимально возможное количество квадратов со стороной C (без наложений). Найти количество квадратов, размещенных на прямоугольнике. Операции умножения и деления не использовать.

Вариант 30.
1. Дано целое число N (>0). Найти значение выражения 1.1 - 1.2 + 1.3 - ... (N слагаемых, знаки чередуются). Условный оператор не использовать.
2. Дано целое число N (>0). С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа N цифра «2». Если имеется, то вывести TRUE, если нет — вывести FALSE.

Вариант 31.

1. Дано вещественное число A и целое число N (>0). Найти A в степени N: A^N = AA ... •A (числа A перемножаются N раз).
2. Дано целое число N (>0). С помощью операций деления нацело и взятия остатка от деления определить, имеются ли в записи числа N нечетные цифры. Если имеются, то вывести TRUE, если нет — вывести FALSE.

Вариант 32.

1. Дано вещественное число A и целое число N (>0). Используя один цикл, вывести все целые степени числа A от 1 до N.
2. Дано целое число N (>0). Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.

Вариант 33.

1. Дано вещественное число A и целое число N (>0). Используя один цикл, найти сумму $1 + A + A^2 + A^3 + ... + A^N$.
2. Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти число, полученное при прочтении числа N справа налево.

Критерии оценивания:

Оценка «5» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Практическое занятие №4

Составление программ с функциями на Python

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с функциями в IDE PyCharm Community.

Пример решения задачи

```python
def rectangle():
    # Вычисление площади прямоугольника
    a = float(input("Ширина %s: " % figure)) # обращение к глобальной
    b = float(input("Высота %s: " % figure)) # переменной figure
    print("Площадь: %.2f" % (a*b))

def triangle():
    # Вычисление площади треугольника
    # Используется общепринятая формула

    a = float(input("Основание %s: " % figure))
    h = float(input("Высота %s: " % figure))
    print("Площадь: %.2f" % (0.5 * a * h))

figure = input("1-прямоугольник, 2-треугольник: ")
if figure == '1':
    rectangle()
elif figure == '2':
    triangle()
```

Варианты заданий

Пояснения.

Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Вариант 1

1. Составить функцию решения задачи: из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится нуль?
2. Описать функцию PowerA234(параметры), вычисляющую вторую, третью и четвертую степень числа A и возвращающую эти степени соответственно в переменные B, C и D. С помощью этой функции найти вторую, третью и четвертую степень пяти данных чисел.

Вариант 2

1. Составить функцию, которая напечатает сорок любых символов.
2. Описать функцию Mean(параметры), вычисляющую среднее арифметическое AMean = (X+Y)/2 и среднее геометрическое GMean = y/X Y двух положительных чисел X и Y. С помощью этой функции найти среднее арифметическое и среднее геометрическое для пар (A, B), (A, C), (A, D), если даны A, B, C, D.

Вариант 3

1. Составить функцию, которая выполнит суммирования числового ряда.

2. Описать функцию TrianglePS(параметры), вычисляющую по стороне a равностороннего треугольника его периметр P = 3*a и площадь S = a2 √3/4. С помощью этой функции найти периметры и площади трех равносторонних треугольников с данными сторонами.

Вариант 4

1. Найти сумму чисел ряда 1,2,3,...,60 с использованием функции нахождения суммы. Использовать локальные переменные.
2. Описать функцию RectPS(x1,y1,x2,y2,P,S), вычисляющую периметр P и площадь S прямоугольника со сторонами, параллельными осям координат, по координатам (x1, y1), (x2, y2) его противоположных вершин (x1, y1, x2, y2 — входные, P и S — выходные параметры вещественного типа). С помощью этой функции найти периметры и площади трех прямоугольников с данными противоположными вершинами.

Вариант 5

1. С помощью функций получить вертикальную и горизонтальную линии. Линия проводится многократной печатью символа. Заключить слово в рамку из полученных линий.
2. Описать функцию DigitCountSum(K, C, S), находящую количество C цифр целого положительного числа K, а также их сумму S (K — входной, C и S — выходные параметры целого типа). С помощью этой функции найти количество и сумму цифр для каждого из пяти данных целых чисел.

Вариант 6

1. Составить функцию, которая выведет на экран строку, содержащую задаваемое с клавиатуры число символов.
2. Описать функцию InvertDigits(K), меняющую порядок следования цифр целого положительного числа K на обратный (K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел.

Вариант7

1. Найти сумму чисел ряда 1,2,3,4,... от числа n до числа m. Суммирование оформить функцией с параметрами. Значения n и m программа должна запрашивать.
2. Описать функцию AddRightDigit(D, K), добавляющую к целому положительному числу K справа цифру D (D — входной параметр целого типа, лежащий в диапазоне 0-9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции последовательно добавить к данному числу K справа данные цифры D1 и D2, выводя результат каждого добавления.

Вариант 8

1. Составить программу, в которой функцию построит изображение, в котором в первой строке 1 звездочка, во второй - 2, в третьей -3, ..., в строке с номером m - m звездочек.
2. Описать функцию AddLeftDigit(D, K), добавляющую к целому положительному числу K слева цифру D (D — входной параметр целого типа, лежащий в диапазоне 1-9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции последовательно добавить к данному числу K слева данные цифры D1 и D2, выводя результат каждого добавления.

Вариант 9

1. Составить программу, в которой функция генерирует четырехзначное число и определяет, есть ли в числе одинаковые цифры.
2. Описать функцию Swap(X, Y), меняющую содержимое переменных X и Y (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных A, B, C, D последовательно поменять содержимое следующих пар: A и B, C и D, B и C и вывести новые значения A, B, C, D.

Вариант 10

1. Составить функцию решения задачи: из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится нуль?
2. Описать функцию Minmax(X, Y), записывающую в переменную X минимальное из значений X и Y, а в переменную Y — максимальное из этих значений (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). Используя четыре вызова этой функции, найти минимальное и максимальное из данных чисел A, B, C, D.

Вариант 11

1. Составить функцию, которая напечатает сорок любых символов.
2. Описать функцию SortInc3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по возрастанию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции упорядочить по возрастанию два данных набора из трех чисел: (Ai, Bi, Ci) и (A2, B2, C2).

Вариант 12

1. Составить функцию, которая выполнит суммирования числового ряда.
2. Описать функцию SortDec3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по убыванию (A, B, C — вещественные параметры, являющиеся одновременно входными и

выходными). С помощью этой функции упорядочить по убыванию два данных набора из трех чисел: (A1, B1, C1) и (A2, B2, C2).

Вариант 13

1. Найти сумму чисел ряда 1,2,3,...,60 с использованием функции нахождения суммы. Использовать локальные переменные.
2. Описать функцию ShiftRight3(A, B, C), выполняющую правый циклический сдвиг: значение A переходит в B, значение B — в C, значение C — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции выполнить правый циклический сдвиг для двух данных наборов из трех чисел: (A1, B1, C1) и (A2, B2, C2).

Вариант 14

1. С помощью функций получить вертикальную и горизонтальную линии. Линия проводится многократной печатью символа. Заключить слово в рамку из полученных линий.
2. Описать функцию ShiftLeft3(A, B, C), выполняющую левый циклический сдвиг: значение A переходит в C, значение C — в B, значение B — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции выполнить левый циклический сдвиг для двух данных наборов из трех чисел: (A1, B1, C1) и (A2, B2, C2).

Вариант 15

1. Составить функцию, которая выведет на экран строку, содержащую задаваемое с клавиатуры число символов
2. Дан прямоугольник, длины сторон которого равны натуральным числам A и B. Составить функцию, которая будет находить на сколько квадратов можно разрезать данный прямоугольник, если от него каждый раз отрезать квадрат наибольшей площади.

Вариант 16

1. Найти сумму чисел ряда 1,2,3,4,... от числа n до числа m. Суммирование оформить функцией с параметрами. Значения n и m программа должна запрашивать.
2. Описать функцию Power1(A, B) вещественного типа, находящую величину AB по формуле AB = exp(B*ln(A)) (параметры A и B — вещественные). В случае нулевого или отрицательного параметра A функция возвращает 0. С помощью этой функции найти степени A^P, B^P, C^P, если даны числа P, A, B, C.

Вариант 17

1. Составить программу, в которой функция построит изображение, в котором в первой строке 1 звездочка, во второй - 2, в третьей -3, ..., в строке с номером m - m звездочек.
2. Описать функцию PowerA3(A, B), вычисляющую третью степень числа A и возвращающую ее в переменную B (A — входной, B — выходной параметр; оба

параметра являются вещественными). С помощью этой функции найти третьи степени пяти данных чисел.

Вариант 18

1. Составить программу, в которой функция генерирует четырехзначное число и определяет, есть ли в числе одинаковые цифры.
2. Описать функцию PowerA234(A, B, C, D), вычисляющую вторую, третью и четвертую степень числа A и возвращающую эти степени соответственно в переменные B, C и D (A — входной, B, C, D — выходные параметры; все параметры являются вещественными). С помощью этой функции найти вторую, третью и четвертую степень пяти данных чисел.

Вариант 19

1. Составить функцию решения задачи: из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится нуль?
2. Описать функцию ShiftLeft3(A, B, C), выполняющую левый циклический сдвиг: значение A переходит в C, значение C — в B, значение B — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции выполнить левый циклический сдвиг для двух данных наборов из трех чисел: (A1, B1, C1) и (A2, B2, C2).

Вариант 20

1. Составить функцию, которая напечатает сорок любых символов.
2. Дан прямоугольник, длины сторон которого равны натуральным числам A и B. Составить функцию, которая будет находить на сколько квадратов можно разрезать данный прямоугольник, если от него каждый раз отрезать квадрат наибольшей площади.

Вариант 21

1. Составить функцию, которая выполнит суммирования числового ряда.
2. Описать функцию Power1(A, B) вещественного типа, находящую величину AB по формуле AB = exp(B*ln(A)) (параметры A и B — вещественные). В случае нулевого или отрицательного параметра A функция возвращает 0. С помощью этой функции найти степени A^P, B^P, C^P, если даны числа P, A, B, C.

Вариант 22

1. Найти сумму чисел ряда 1,2,3,...,60 с использованием функции нахождения суммы. Использовать локальные переменные.
2. Описать функцию Mean(X, Y, AMean, GMean), вычисляющую среднее арифметическое AMean = (X+Y)/2 и среднее геометрическое GMean = y/X Y двух положительных чисел X и Y (X и Y — входные, AMean и GMean — выходные параметры вещественного типа). С помощью этой функции найти среднее

арифметическое и среднее геометрическое для пар (A, B), (A, C), (A, D), если даны A, B, C, D.

Вариант 23

1. С помощью функций получить вертикальную и горизонтальную линии. Линия проводится многократной печатью символа. Заключить слово в рамку из полученных линий.
2. Описать функцию TrianglePS(a, P, S), вычисляющую по стороне a равностороннего треугольника его периметр P = 3*a и площадь S = a2 √3/4 (a — входной, P и S — выходные параметры; все параметры являются вещественными). С помощью этой функции найти периметры и площади трех равносторонних треугольников с данными сторонами.

Вариант 24

1. Составить функцию, которая выведет на экран строку, содержащую задаваемое с клавиатуры число символов.
2. Описать функцию RectPS(x1,y1,x2,y2,P, S), вычисляющую периметр P и площадь S прямоугольника со сторонами, параллельными осям координат, по координатам (x1, y1), (x2, y2) его противоположных вершин (x1, y1, x2, y2 — входные, P и S — выходные параметры вещественного типа). С помощью этой функции найти периметры и площади трех прямоугольников с данными противоположными вершинами.

Вариант 25

1. Найти сумму чисел ряда 1,2,3,4,... от числа n до числа m. Суммирование оформить функцией с параметрами. Значения n и m программа должна запрашивать.
2. Описать функцию DigitCountSum(K, C, S), находящую количество C цифр целого положительного числа K, а также их сумму S (K — входной, C и S — выходные параметры целого типа). С помощью этой функции найти количество и сумму цифр для каждого из пяти данных целых чисел.

Вариант 26

1. Составить программу, в которой функцию построит изображение, в котором в первой строке 1 звездочка, во второй - 2, в третьей -3, ..., в строке с номером m - m звездочек.
2. Описать функцию InvertDigits(K), меняющую порядок следования цифр целого положительного числа K на обратный (K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел.

Вариант 27

1. Составить программу, в которой функция генерирует четырехзначное число и определяет, есть ли в числе одинаковые цифры.
2. Описать функцию AddRightDigit(D, K), добавляющую к целому положительному числу K справа цифру D (D — входной параметр целого типа, лежащий в диапазоне 0-9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции последовательно добавить к данному числу K справа данные цифры D1 и D2, выводя результат каждого добавления.

Вариант 28

1. Составить функцию решения задачи: из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится нуль?
2. Описать функцию AddLeftDigit(D, K), добавляющую к целому положительному числу K слева цифру D (D — входной параметр целого типа, лежащий в диапазоне 1-9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции последовательно добавить к данному числу K слева данные цифры D1 и D2, выводя результат каждого добавления.

Вариант 29

1. Составить функцию, которая напечатает сорок любых символов.
2. Описать функцию Swap(X, Y), меняющую содержимое переменных X и Y (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных A, B, C, D последовательно поменять содержимое следующих пар: A и B, C и D, B и C и вывести новые значения A, B, C, D.

Вариант 30

1. Составить функцию, которая выполнит суммирования числового ряда.
2. Описать функцию Minmax(X, Y), записывающую в переменную X минимальное из значений X и Y, а в переменную Y — максимальное из этих значений (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). Используя четыре вызова этой функции, найти минимальное и максимальное из данных чисел A, B, C, D.

Вариант 31

1. Найти сумму чисел ряда 1,2,3,...,60 с использованием функции нахождения суммы. Использовать локальные переменные.
2. Описать функцию SortInc3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по возрастанию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции упорядочить по возрастанию два данных набора из трех чисел: (A_i, B_i, C_i) и (A_2, B_2, C_2).

Вариант 32

1. С помощью функций получить вертикальную и горизонтальную линии. Линия проводится многократной печатью символа. Заключить слово в рамку из полученных линий.
2. Описать функцию SortDec3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по убыванию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции упорядочить по убыванию два данных набора из трех чисел: (A_1, B_1, C_1) и (A_2, B_2, C_2).

Вариант 33

1. Составить функцию, которая выведет на экран строку, содержащую задаваемое с клавиатуры число символов.
2. Описать функцию ShiftRight3(A, B, C), выполняющую правый циклический сдвиг: значение A переходит в B, значение B — в C, значение C — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой функции выполнить правый циклический сдвиг для двух данных наборов из трех чисел: (A1, B1, C1) и (A2, B2, C2).

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет предоставлены, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет предоставлены, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Практическое занятие №5

Составление программ со списками и строками на Python

Цели закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ со списками в IDE PyCharm Community.

Пример решения задачи

```
# Программа запрашивает с клавиатуры пять чисел, добавляет их в список.
# На экран выводит их сумму, максимальное и минимальное из них.

ListAppend = []
i = 0
while i < 5:
    ListAppend.append(int(input('Введи значение списка:  ')))
    i += 1
print(ListAppend)
print('Сумма элементов списка:   ', sum(ListAppend))
print('Минимальный элемент списка:   ', min(ListAppend))
print('Максимальный элемент списка:   ', max(ListAppend))
```

Пояснения.

Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Разместить на GitHub проект и отчет по практическому занятию.

Варианты заданий

Вариант 1

1. Сформировать и вывести целочисленный список размера 10, содержащий 10 первых положительных нечетных чисел: 1,3,5, ...

2. Дан список размера N. Найти номера тех элементов списка, которые больше своего правого соседа, и количество таких элементов. Найденные номера выводить в порядке их возрастания.

3. Дан список A размера N и целые числа K и L ($1 < K < L < N$). Переставить в обратном порядке элементы списка, расположенные между элементами A_K и A_L, не включая эти элементы.

Вариант 2

1. Сформировать и вывести целочисленный список размера 10, содержащий степени двойки от первой до 10-й: 2, 4, 8,16, ...

2. Дан список размера N. Найти номера тех элементов список, которые больше своего левого соседа, и количество таких элементов. Найденные номера выводить в порядке их убывания.

3. Дан список размера N. Обнулить элементы списка, расположенные между его минимальным и максимальным элементами (не включая минимальный и максимальный элементы).

Вариант 3

1. Дан первый член A и разность D арифметической прогрессии. Сформировать и вывести список размера 10, содержащий 10 первых членов данной прогрессии: A, A + D, A + 2*D, A + 3*D, ...

2. Дан список размера N. Найти номер его первого локального минимума (локальный минимум — это элемент, который меньше любого из своих соседей).

3. Дан список размера N (N — четное число). Поменять местами его первый элемент со вторым, третий — с четвертым и т. д.

Вариант 4

1. Дан первый член A и знаменатель D геометрической прогрессии. Сформировать и вывести список размера 10, содержащий 10 первых членов данной прогрессии: A, A* D, A* D^2, A*D^3, ...

2. Дан список размера N. Найти номер его последнего локального максимума (локальный максимум — это элемент, который больше любого из своих соседей).

3. Дан список размера N. Переставить в обратном порядке элементы список, расположенные между его минимальным и максимальным элементами, включая минимальный и максимальный элементы.

Вариант 5

1. Дано целое число N (>2). Сформировать и вывести целочисленный список размера 10, содержащий 10 первых элементов последовательности чисел Фибоначчи F_K: $F_1 = 1$, $F_2 = 1$, $F_K = F_{K-2} + F_{K-1}$, K = 3,4,... .

2. Дан список A размера N и целые числа K и L ($1 < K < L < N$). Переставить в обратном порядке элементы списка, расположенные между элементами A_K и A_L, включая эти элементы.

3. Дан список размера N. Обнулить все его локальные максимумы (то есть числа, большие своих соседей).

Вариант 6

1. Даны целые числа N (>2), A и B. Сформировать и вывести целочисленный список размера 10, первый элемент которого равен A, второй равен B, а каждый последующий элемент равен сумме всех предыдущих.

2. Дан список размера N. Найти максимальный из его локальных минимумов (локальный минимум — это элемент, который меньше любого из своих соседей).

3. Дан список размера N и целое число K ($1 < K < N$). Осуществить сдвиг элементов список вправо на K позиций (при этом A_1 перейдет в A_{K+1}, A_2 — в A_{K+2}, ..A_{N-K} —

в A_N, а исходное значение K последних элементов будет потеряно). Первые K элементов полученного списка положить равными 0.

Вариант 7

1. Дан целочисленный список размера 10. Вывести все содержащиеся в данном списке нечетные числа в порядке возрастания их индексов, а также их количество K.

2. Дан список размера N. Найти минимальный из его локальных максимумов (локальный минимум — это элемент, который меньше любого из своих соседей).

3. Дан список размера N. Возвести в квадрат все его локальные минимумы (то есть числа, меньшие своих соседей).

Вариант 8

1. Дан целочисленный список размера 10. Вывести все содержащиеся в данном списе четные числа в порядке убывания их индексов, а также их количество K.

2. Дан список размера N. Найти количество участков, на которых его элементы монотонно возрастают.

3. Дан список размера N. Заменить каждый элемент списка на среднее арифметическое этого элемента и его соседей.

Вариант 9

1. Дан целочисленный список размера 10. Вывести вначале все содержащиеся в данном списке четные числа в порядке возрастания их индексов, а затем — все нечетные числа в порядке убывания их индексов.

2. Дан список размера N. Найти количество участков, на которых его элементы монотонно убывают.

3. Дано множество A из N точек на плоскости и точка B (точки заданы своими координатами x, y). Найти точку из множества A, наиболее близкую к точке B. Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два список: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 10

1. Дан список A размера N и целое число K ($1 < K < N$). Вывести элементы список с порядковыми номерами, кратными K: A_K, A_{2*K}, A_{3*K},... . Условный оператор не использовать.

2. Дан список размера N. Найти количество его промежутков монотонности (то есть участков, на которых его элементы возрастают или убывают).

3. Дан список размера N. Осуществить сдвиг элементов список вправо на одну позицию (при этом A_1 перейдет в A_2, A_2 — в A_3, ..., A_{N-1} — в A_N, а исходное значение последнего элемента будет потеряно). Первый элемент полученного списка положить равным 0.

Вариант 11

1. Дан список A размера N (N — четное число). Вывести его элементы с четными номерами в порядке возрастания номеров: A_2, A_4, A_6, ..., A_N. Условный оператор не использовать.

2. Дано число R и список A размера N. Найти элемент списка, который наиболее близок к числу R (то есть такой элемент A_K, для которого величина $|A_K - R|$ является минимальной).

3. Дан список размера N. Осуществить сдвиг элементов списка влево на одну позицию (при этом A_N перейдет в A_{N-1}, A_{N-1} — в A_{N-2}, .., A_2 — в A_1, а исходное значение первого элемента будет потеряно). Последний элемент полученного списка положить равным 0.

Вариант 12

1. Дан список A размера N (N — нечетное число). Вывести его элементы с нечетными номерами в порядке убывания номеров: A_N, A_{N-2}, A_{N-4}, ..., A_1. Условный оператор не использовать.

2. Дан список размера N. Найти два соседних элемента, сумма которых максимальна, и вывести эти элементы в порядке возрастания их индексов.

3. Дан список размера N и целое число K ($1 < K < N$). Осуществить сдвиг элементов списка вправо на K позиций (при этом A_1 перейдет в A_{K+1}, A_2 — в A_{K+2}, ..A_{N-K} — в A_N, а исходное значение K последних элементов будет потеряно). Первые K элементов полученного списка положить равными 0.

Вариант 13

1. Дан список A размера N. Вывести вначале его элементы с четными номерами (в порядке возрастания номеров), а затем — элементы с нечетными номерами (также в порядке возрастания номеров): A_2, A_4, A_6, . . ., A_1, A_3, A_5, ... Условный оператор не использовать.

2. Дано число R и список размера N. Найти два соседних элемента списка, сумма которых наиболее близка к числу R, и вывести эти элементы в порядке возрастания их индексов (определение наиболее близких чисел - то есть такой элемент A_K, для которого величина $|A_K - R|$ является минимальной).

3. Дан список размера N и целое число K ($1 < K < N$). Осуществить сдвиг элементов списка влево на K позиций (при этом A_N перейдет в A_{N-K}, A_{N-1} — в A_{N-K-1}, ..A_{K+1} —

в A_1, а исходное значение K первых элементов будет потеряно). Последние K элементов полученного списка положить равными 0.

Вариант 14

1. Дан список A размера N. Вывести вначале его элементы с нечетными номерами в порядке возрастания номеров, а затем — элементы с четными номерами в порядке убывания номеров: A_1, A_3, A_5, …, A_6, A_4, A_2. Условный оператор не использовать.

2. Дан список A размера N. Сформировать два новых списка B и C: в список B записать все положительные элементы список A, в список C — все отрицательные (сохраняя исходный порядок следования элементов). Вывести вначале размер и содержимое списка B, а затем — размер и содержимое списка C.

3. Дан список размера N. Осуществить циклический сдвиг элементов списка вправо на одну позицию (при этом A_1 перейдет в A_2, A_2 — в A_3,..., A_N — в A_1).

Вариант 15

1. Дан список A размера N. Вывести его элементы в следующем порядке: A_1, A_N, A_2, A_{N-1}, A_3, A_{N-2}, ….

2. Дан целочисленный список размера N, все элементы которого упорядочены (по возрастанию или по убыванию). Найти количество различных элементов в данном списке.

3. Дан список размера N. Осуществить циклический сдвиг элементов списка влево на одну позицию (при этом A_N перейдет в A_{N-1}, A_{N-1} — в A_{N-2}, . . ., A_1 — в A_N).

Вариант 16

1. Дан список A размера N. Вывести его элементы в следующем порядке: A_1, A_2, A_N, A_{N-1}, A_3, A_4, A_{N-2}, A_{N-3}, … .

2. Дан целочисленный список размера N, содержащий ровно два одинаковых элемента. Найти номера одинаковых элементов и вывести эти номера в порядке возрастания.

3. Дан список A размера N и целое число K ($1 < K < 4$, $K < N$). Осуществить циклический сдвиг элементов списка вправо на K позиций (при этом A_1 перейдет в A_{K+1}, A_2 — в A_{K+2}, ..., A_N — в A_K). Допускается использовать вспомогательный список из 4 элементов.

Вариант 17

1. Дан список A ненулевых целых чисел размера 10. Вывести значение первого из тех его элементов A_K, которые удовлетворяют неравенству $A_K < A_{10}$. Если таких элементов нет, то вывести 0.

2. Дан список размера N. Найти номера двух ближайших элементов из этого списка (то есть элементов с наименьшим модулем разности) и вывести эти номера в порядке возрастания.

3. Дан список A размера N и целое число K ($1 < K < 4$, $K < N$). Осуществить циклический сдвиг элементов списка влево на K позиций (при этом A_N перейдет в A_{N_K}, A_{N-1} — в A_{N-K-1}, ..., A_1 — в A_{N-K+1}). Допускается использовать вспомогательный список из 4 элементов.

Вариант 18

1. Дан целочисленный список A размера 10. Вывести порядковый номер последнего из тех его элементов A_K, которые удовлетворяют двойному неравенству $A_1 < A_K < A_{10}$. Если таких элементов нет, то вывести 0.

2. Дано число R и список размера N. Найти два различных элемента списка, сумма которых наиболее близка к числу R, и вывести эти элементы в порядке возрастания их индексов (определение наиболее близких чисел - то есть такой элемент A_K, для которого величина $|A_K - R|$ является минимальной).

3. Дан список размера N, все элементы которого, кроме первого, упорядочены по возрастанию. Сделать список упорядоченным, переместив первый элемент на новую позицию.

Вариант 19

1. Дан список размера N и целые числа K и L ($1 < K < L < N$). Найти сумму элементов списка с номерами от K до L включительно.

2. Дан целочисленный список размера N. Найти количество различных элементов в данном списке.

3. Дан список размера N, все элементы которого, кроме последнего, упорядочены по возрастанию. Сделать список упорядоченным, переместив последний элемент на новую позицию.

Вариант 20

1. Дан список размера N и целые числа K и L ($1 < K < L < N$). Найти среднее арифметическое элементов список с номерами от K до L включительно.

2. Дан целочисленный список размера N. Найти максимальное количество его одинаковых элементов.

3. Дан список размера N, все элементы которого, кроме одного, упорядочены по убыванию. Сделать список упорядоченным, переместив элемент, нарушающий упорядоченность, на новую позицию.

Вариант 21

1. Дан список размера N и целые числа K и L (1 < K < L < N). Найти сумму всех элементов списка, кроме элементов с номерами от K до L включительно.

2. Дан целочисленный список размера N. Если он является перестановкой, то есть содержит все числа от 1 до N, то вывести 0; в противном случае вывести номер первого недопустимого элемента.

3. Дано множество A из N точек на плоскости и точка B (точки заданы своими координатами x, y). Найти точку из множества A, наиболее близкую к точке B. Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант22

1. Дан список размера N и целые числа K и L (1 < K ≤ L ≤ N). Найти среднее арифметическое всех элементов списка, кроме элементов с номерами от K до L включительно.

2. Даны списки A и B одинакового размера N. Поменять местами их содержимое и вывести вначале элементы преобразованного списка A, а затем — элементы преобразованного списка B.

3. Дано множество A из N точек (точки заданы своими координатами x, y). Среди всех точек этого множества, лежащих во второй четверти, найти точку, наиболее удаленную от начала координат. Если таких точек нет, то вывести точку с нулевыми координатами.

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 23

1. Дан целочисленный список размера N, не содержащий одинаковых чисел. Проверить, образуют ли его элементы арифметическую прогрессию. Если образуют, то вывести разность прогрессии, если нет — вывести 0.

2. Дан список A размера N. Сформировать новый список B того же размера, элементы которого определяются следующим образом:

   $B_K = 2*A_K$, если $A_K < 5$,

   $A_K/2$ в противном случае.

3. Дано множество A из N точек (точки заданы своими координатами x, y). Среди всех точек этого множества, лежащих в первой или третьей четверти, найти точку, наиболее близкую к началу координат. Если таких точек нет, то вывести точку с нулевыми координатами.

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 24

1. Дан список ненулевых целых чисел размера N. Проверить, образуют ли его элементы геометрическую прогрессию. Если образуют, то вывести знаменатель прогрессии, если нет — вывести 0.

2. Дан целочисленный список A размера N. Переписать в новый целочисленный список B того же размера вначале все элементы исходного списка с четными номерами, а затем — с нечетными: A_2, A_4, A_6, …, A_1, A_3, A_5, … . Условный оператор не использовать.

3. Дано множество A из N точек (точки заданы своими координатами x, y). Найти пару различных точек этого множества с максимальным расстоянием между ними и само это расстояние (точки выводятся в том же порядке, в котором они перечислены при задании множества A).

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 25

1. Дан целочисленный список размера N. Проверить, чередуются ли в нем четные и нечетные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.

2. Даны два списка A и B одинакового размера N. Сформировать новый список C того же размера, каждый элемент которого равен максимальному из элементов списков A и B.

3. Даны множества A и B, состоящие соответственно из N1 и N2 точек (точки заданы своими координатами x, y). Найти минимальное расстояние между точками этих множеств и сами точки, расположенные на этом расстоянии (вначале выводится точка из множества A, затем точка из множества B).

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 26

1. Дан список ненулевых целых чисел размера N. Проверить, чередуются ли в нем положительные и отрицательные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.

2. Дан целочисленный список A размера N (< 15). Переписать в новый целочисленный список B все элементы с нечетными порядковыми номерами (1,3,...) и вывести размер полученного списка B и его содержимое. Условный оператор не использовать.

3. Дано множество A из N точек (N > 2, точки заданы своими координатами x, y). Найти такую точку из данного множества, сумма расстояний от которой до остальных его точек минимальна, и саму эту сумму.

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 27

1. Дан список A размера N. Найти минимальный элемент из его элементов с четными номерами: A_2, A_4, A_6, ...

2. Дан целочисленный список A размера N. Переписать в новый целочисленный список B все четные числа из исходного списка (в том же порядке) и вывести размер полученного список B и его содержимое.

3. Дано множество A из N точек (N > 2, точки заданы своими координатами x, y). Найти наибольший периметр треугольника, вершины которого принадлежат различным точкам множества A, и сами эти точки (точки выводятся в том же порядке, в котором они перечислены при задании множества A).

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два список: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 28

1. Дан список A размера N. Найти максимальный элемент из его элементов с нечетными номерами: A_1, A_3, A_5, ...

2. Дан целочисленный список A размера N (< 15). Переписать в новый целочисленный список B все элементы с порядковыми номерами, кратными трем (3, 6, ...), и вывести размер полученного списка B и его содержимое. Условный оператор не использовать.

3. Дано множество A из N точек (N > 2, точки заданы своими координатами x, y). Найти наименьший периметр треугольника, вершины которого принадлежат различным точкам множества A, и сами эти точки (точки выводятся в том же порядке, в котором они перечислены при задании множества A).

Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле:

$$R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}.$$

Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Вариант 29

1. Дан список A размера N и целое число K (1 < K < N). Преобразовать список, увеличив каждый его элемент на исходное значение элемента A_K.

2. Дан целочисленный список A размера N. Переписать в новый целочисленный список B того же размера вначале все элементы исходного списка с четными номерами, а затем — с нечетными: A_2, A_4, A_6, …, A_1, A_3, A_5, … . Условный оператор не использовать.

3. Дан список размера N. Осуществить сдвиг элементов списка вправо на одну позицию (при этом A_1 перейдет в A_2, A_2 — в A_3, ..., A_{N-1} — в A_N, а исходное значение последнего элемента будет потеряно). Первый элемент полученного списка положить равным 0.

Вариант 30

1. Дан целочисленный список размера N. Увеличить все четные числа, содержащиеся в списке, на исходное значение первого четного числа. Если четные числа в списке отсутствуют, то оставить список без изменений.

2. Дан список A размера N. Сформировать новый список B того же размера по следующему правилу: элемент B_K равен сумме элементов списка A с номерами от 1 до K.

3. Дан список размера N и целое число K (1 < K < N). Осуществить сдвиг элементов списка вправо на K позиций (при этом A_1 перейдет в A_{K+1}, A_2 — в A_{K+2}, ..A_{N-K} — в A_N, а исходное значение K последних элементов будет потеряно). Первые K элементов полученного списка положить равными 0.

Вариант 31

1. Дан целочисленный список размера N. Увеличить все нечетные числа, содержащиеся в списке, на исходное значение последнего нечетного числа. Если нечетные числа в списке отсутствуют, то оставить список без изменений.

2. Дан список A размера N. Сформировать новый список B того же размера по следующему правилу: элемент B_K равен среднему арифметическому элементов списка A с номерами от 1 до K.

3. Дан список размера N, все элементы которого, кроме последнего, упорядочены по возрастанию. Сделать список упорядоченным, переместив последний элемент на новую позицию.

Вариант 32

1. Дан список размера N. Поменять местами его минимальный и максимальный элементы.

2. Дан список A размера N. Сформировать новый список B того же размера по следующему правилу: элемент B_K равен сумме элементов списка A с номерами от K до N.

3. Дан список размера N. Осуществить циклический сдвиг элементов списка вправо на одну позицию (при этом A_1 перейдет в A_2, A_2 — в A_3,..., A_N — в A_1).

Вариант 33

1. Дан список размера N (N — четное число). Поменять местами первую и вторую половины списка.

2. Дан список A размера N. Сформировать новый список B того же размера по следующему правилу: элемент B_K равен среднему арифметическому элементов списка A с номерами от K до N.

3. Дан список размера N и целое число K ($1 < K < N$). Осуществить сдвиг элементов списка влево на K позиций (при этом A_N перейдет в A_{N-K}, A_{N-1} — в A_{N-K-1}, ..A_{K+1} — в A_1, а исходное значение K первых элементов будет потеряно). Последние K элементов полученного списка положить равными 0.

Дополнительные задачи

Даны два списка A и B размера 5, элементы которых упорядочены по возрастанию. Объединить эти списки так, чтобы результирующий список C (размера 10) остался упорядоченным по возрастанию.

Даны три целочисленных списка A, B и C размера N_A, N_B, N_C соответственно, элементы которых упорядочены по убыванию. Объединить эти списки так, чтобы результирующий целочисленный список D (размера $N_A + N_B + N_C$) остался упорядоченным по убыванию.

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Практическое занятие №6

Составление программ со словарями и множествами на Python

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ со строками в IDE PyCharm Community.

Пример решения задачи

```
# В заданной строке найти все прописные буквы, посчитать их количество.
# Использовать библиотеку string

from string import ascii_uppercase

string_new = 'In PyCharm, you can specify third-party standalone applications and run them as External Tools'
str_1 = [i for i in string_new if i in ascii_uppercase]
print(len(str_1))
print(str_1)
```

Пояснения.

Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Разместить на GitHub проект и отчет по практическому занятию.

Варианты заданий

Вариант 1

1. Дан символ C. Вывести его код (то есть номер в кодовой таблице).
2. Дана строка, содержащая латинские буквы и скобки трех видов: «()», «[]», «{}». Если скобки расставлены правильно (то есть каждой открывающей соответствует закрывающая скобка того же вида), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная скобка, или, если закрывающих скобок не хватает, число —1.

Вариант 2

1. Дано целое число N (32 < N < 126). Вывести символ с кодом, равным N.
2. Дана строка, содержащая латинские буквы и круглые скобки. Если скобки расставлены правильно (то есть каждой открывающей соответствует одна закрывающая), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная закрывающая скобка, или, если закрывающих скобок не хватает, число —1.

Вариант 3

1. Дан символ C. Вывести два символа, первый из которых предшествует символу C в кодовой таблице, а второй следует за символом C.
2. Дана строка, содержащая цифры и строчные латинские буквы. Если буквы в строке упорядочены по алфавиту, то вывести 0; в противном случае вывести номер первого символа строки, нарушающего алфавитный порядок.

Вариант 4

1. Дано целое число N (1 < N < 26). Вывести N первых прописных (то есть заглавных) букв латинского алфавита.

2. Дана строка-предложение. Зашифровать ее, поместив вначале все символы, расположенные на четных позициях строки, а затем, в обратном порядке, все символы, расположенные на нечетных позициях (например, строка «Программа» превратится в «ргамамроП»).

Вариант 5

1. Дано целое число N (1 < N < 26). Вывести N последних строчных (то есть маленьких) букв латинского алфавита в обратном порядке (начиная с буквы «z»).
2. Дана строка-предложение на русском языке и число K (0 < K < 10). Зашифровать строку, выполнив циклическую замену каждой буквы на букву того же регистра, расположенную в алфавите на K-й позиции после шифруемой буквы (например, для K = 2 «А» перейдет в «В», «а» — в «в», «Б» — в «Г», «я» — в «б» и т. д.). Букву «ё» в алфавите не учитывать, знаки препинания и пробелы не изменять.

Вариант 6

1. Дан символ C, изображающий цифру или букву (латинскую или русскую). Если C изображает цифру, то вывести строку «digit», если латинскую букву — вывести строку «lat», если русскую — вывести строку «rus».
2. Дана строка-предложение на русском языке. Зашифровать ее, выполнив циклическую замену каждой буквы на следующую за ней в алфавите и сохранив при этом регистр букв («А» перейдет в «Б», «а» — в «б», «Б» — в «В», «я» — в «а» и т. д.). Букву «ё» в алфавите не учитывать («е» должна переходить в «ж»). Знаки препинания и пробелы не изменять.

Вариант 7

1. Дана непустая строка. Вывести коды ее первого и последнего символа.
2. Дана строка, содержащая полное имя файла, то есть имя диска, список каталогов (путь), собственно имя и расширение. Выделить из этой строки расширение файла (без предшествующей точки).

Вариант 8

1. Дано целое число N (>0) и символ C. Вывести строку длины N, которая состоит из символов C.
2. Дана строка-предложение с избыточными пробелами между словами. Преобразовать ее так, чтобы между словами был ровно один пробел.

Вариант 9

1. Дано четное число N (>0) и символы C_1 и C_2. Вывести строку длины N, которая состоит из чередующихся символов C_1 и C_2, начиная с C_1.
2. Дана строка, содержащая полное имя файла, то есть имя диска, список каталогов (путь), собственно имя и расширение. Выделить из этой строки имя файла (без расширения).

Вариант 10

1. Дана строка. Вывести строку, содержащую те же символы, но расположенные в обратном порядке.

2. Дана строка, содержащая полное имя файла. Выделить из этой строки название последнего каталога (без символов «\»). Если файл содержится в корневом каталоге, то вывести символ «\ ».

Вариант 11

1. Дана непустая строка S. Вывести строку, содержащую символы строки S, между которыми вставлено по одному пробелу.
2. Дана строка, содержащая полное имя файла. Выделить из этой строки название первого каталога (без символов «\»). Если файл содержится в корневом каталоге, то вывести символ «\».

Вариант 12

1. Дана непустая строка S и целое число N (> 0). Вывести строку, содержащую символы строки S, между которыми вставлено по N символов «*» (звездочка).
2. Дана строка-предложение на русском языке. Вывести самое короткое слово в предложении. Если таких слов несколько, то вывести последнее из них. Словом считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки.

Вариант 13

1. Дана строка. Подсчитать количество содержащихся в ней цифр.
2. Дана строка-предложение на русском языке. Вывести самое длинное слово в предложении. Если таких слов несколько, то вывести первое из них. Словом считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки.

Вариант 14

1. Дана строка. Подсчитать количество содержащихся в ней прописных латинских букв.
2. Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке гласных букв.

Вариант 15

1. Дана строка. Подсчитать общее количество содержащихся в ней строчных латинских и русских букв.
2. Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке знаков препинания.

Вариант 16

1. Дана строка. Преобразовать в ней все прописные латинские буквы в строчные.
2. Дана строка-предложение на русском языке. Преобразовать строку так, чтобы каждое слово начиналось с заглавной буквы. Словом считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки. Слова, не начинающиеся с буквы, не изменять.

Вариант 17

1. Дана строка. Преобразовать в ней все строчные буквы (как латинские, так и русские) в прописные.
2. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним пробелом и расположенные в алфавитном порядке.

Вариант 18

1. Дана строка. Преобразовать в ней все строчные буквы (как латинские, так и русские) в прописные, а прописные — в строчные.
2. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним пробелом и расположенные в обратном порядке.

Вариант 19

1. Дана строка. Если она представляет собой запись целого числа, то вывести 1, если вещественного (с дробной частью) — вывести 2; если строку нельзя преобразовать в число, то вывести 0. Считать, что дробная часть вещественного числа отделяется от его целой части десятичной точкой «.».
2. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все предыдущие вхождения его последней буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «.ИНИ.УМ». Количество пробелов между словами не изменять.

Вариант 20

1. Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке слева направо).
2. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все последующие вхождения его первой буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «МИНИ.У». Количество пробелов между словами не изменять.

Вариант 21

1. Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке справа налево).
2. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним символом «.» (точка). В конце строки точку не ставить.

Вариант 22

1. Дана строка, изображающая целое положительное число. Вывести сумму цифр этого числа.
2. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого длинного слова.

Вариант 23

1. Дана строка, изображающая арифметическое выражение вида «<цифра>±<цифра>±.. .±<цифра>», где на месте знака операции «±» находится символ «+» или «-» (например, «4+7-2—8»). Вывести значение данного выражения (целое число).
2. Даны строки S, S_1 и S_2. Заменить в строке S все вхождения строки S_1 на строку S_2.

Вариант 24

1. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат ровно три буквы «А».
2. Даны строки S, S_1 и S_2. Заменить в строке S первое вхождение строки S_1 на строку S_2.

Вариант 25

1. Дано целое число N (>0) и строка S. Преобразовать строку S в строку длины N следующим образом: если длина строки S больше N, то отбросить первые символы, если длина строки S меньше N, то в ее начало добавить символы «.» (точка).
2. Даны строки S, S_1 и S_2. Заменить в строке S последнее вхождение строки S_1 на строку S_2.

Вариант 26

1. Даны целые положительные числа N1 и N2 и строки S1 и S2. Получить из этих строк новую строку, содержащую первые N1 символов строки S1 и последние N2 символов строки S2 (в указанном порядке).
2. Даны строки S и S_0. Удалить из строки S все подстроки, совпадающие с S_0. Если совпадающих подстрок нет, то вывести строку S без изменений.

Вариант 27

1. Дан символ C и строка S. Удвоить каждое вхождение символа C в строку S.
2. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти количество слов в строке.

Вариант 28

1. Дан символ C и строки S, S_0. Перед каждым вхождением символа C в строку S вставить строку S_0.
2. Дана строка, содержащая по крайней мере один символ пробела. Вывести подстроку, расположенную между первым и последним пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.

Вариант 29

1. Дан символ C и строки S, S_0. После каждого вхождения символа C в строку S вставить строку S_0.
2. Дана строка, содержащая по крайней мере один символ пробела. Вывести подстроку, расположенную между первым и вторым пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.

Вариант 30

1. Даны строки S и S_0. Проверить, содержится ли строка S_0 в строке S. Если содержится, то вывести TRUE, если не содержится, то вывести FALSE.
2. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые начинаются и заканчиваются одной и той же буквой.

Вариант 31

1. Даны строки S и S_0. Найти количество вхождений строки S_0 в строку S.
2. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого короткого слова.

Вариант 32

1. Даны строки S и S_0. Удалить из строки S первую подстроку, совпадающую с S_0. Если совпадающих подстрок нет, то вывести строку S без изменений.
2. Дана строка-предложение с избыточными пробелами между словами. Преобразовать ее так, чтобы между словами был ровно один пробел.

Вариант 33

1. Даны строки S и S_0. Удалить из строки S последнюю подстроку, совпадающую с S_0. Если совпадающих подстрок нет, то вывести строку S без изменений.
2. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат хотя бы одну букву «А».

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Тема 1.3
Файлы

Проверяемые результаты обучения по общим и профессиональным компетенциям

− ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

− ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности

− ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности

- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части

Ответьте на вопросы.

- Понятие файла и объекта файлового типа.
- Последовательность действий при работе с файлами.
- Понятие текстового файла.
- Понятие байтового файла.
- Назначение функции open().
- Назначение файлового метода read().
- Методы для записи в файл.

Устный опрос оценивается:
Оценка «5» – полный развернутый ответ;
Оценка «4» – полный неразвернутый ответ;
Оценка «3» – неполный ответ;
Оценка «2» – ответа нет.

Практическое занятие №7

Составление программ для работы с текстовыми файлами

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыкисоставление программ с текстовыми файлами вIDE PyCharm Community.

Пример решения задачи

```
f1 = open('new_file_1.txt', 'r', encoding="utf-8")
print('Читаем и выводим на экран первые 10 байт или символов')
print(f1.read(30))
f1.close()

f1 = open('new_file_1.txt', encoding="utf-8")
print('Читаем и выводим на экран весь файл')
print(f1.read())
f1.close()

f1 = open('new_file_1.txt', encoding="utf-8")
print('Читаем и выводим на экран весь файл с помощью for')
for line in f1:
    print(line, end='')
print(type(f1.read()))
f1.close()
```

Пояснения.

Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Разместить на GitHub проект и отчет по практическому занятию.

Варианты заданий

Вариант 1.

1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Отрицательные нечетные элементы:
Сумма отрицательных нечетных элементов:
Среднее арифметическое отрицательных нечетных элементов:

2. Из предложенного текстового файла (text18-1.txt) вывести на экран его содержимое, количество букв в верхнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между первой и второй.

Вариант 2.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Максимальный элемент:
Произведение элементов меньших 0 в первой половине:

2. Из предложенного текстового файла (text18-2.txt) вывести на экран его содержимое, количество знаков препинания. Сформировать новый файл, в который поместить текст в стихотворной форме выведя строки в обратном порядке.

Вариант 3.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Минимальный элемент:
Квадраты четных элементов:
Сумма квадратов четных элементов:
Среднее арифметическое суммы квадратов четных элементов:

2. Из предложенного текстового файла (text18-3.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы третей строки их числовыми кодами.

Вариант 4.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Минимальный элемент:
Элементы, умноженные на первый максимальный элемент:

2. Из предложенного текстового файла (text18-4.txt) вывести на экран его содержимое, количество символов, принадлежащих к группе букв. Сформировать новый файл, в

который поместить текст в стихотворной форме предварительно заменив символы верхнего регистра на нижний.

Вариант 5.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Сумма элементов:
Элементы, умноженные на минимальный элемент:

2. Из предложенного текстового файла (text18-5.txt) вывести на экран его содержимое, количество символов в тексте. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы нижнего регистра на верхний.

Вариант 6.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Произведение элементов:
Повторяющиеся элементы:
Количество повторяющихся элементов:
Элементы больше 5 увеличены в два раза:

2. Из предложенного текстового файла (text18-6.txt) вывести на экран его содержимое, количество пробельных символов. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив все знаки пунктуации на знак «!».

Вариант 7.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Среднее арифметическое элементов:
Положительные четные элементы:
Сумма положительных четных элементов:
Среднее арифметическое положительных четных элементов:

2. Из предложенного текстового файла (text18-7.txt) вывести на экран его содержимое, количество букв в нижнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между второй и третьей.

Вариант 8.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Индекс последнего минимального элемента:
Сумма элементов больших 10 во второй половине:

2. Из предложенного текстового файла (text18-8.txt) вывести на экран его содержимое, количество символов, принадлежащих к группе букв. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно удалив букву «с» из текста.

Вариант 9.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Индекс последнего максимального элемента:
Меняем местами первую и последнюю трети:

2. Из предложенного текстового файла (text18-9.txt) вывести на экран его содержимое, количество букв в нижнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку фразой введенной пользователем.

Вариант 10.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Элементы в обратном порядке:

Сумма элементов последней половины:

2. Из предложенного текстового файла (text18-10.txt) вывести на экран его содержимое, количество букв в верхнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив после последней строки автора и название произведения.

Вариант 11.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Минимальный элемент:
Количество положительных элементов в первой половине:

2. Из предложенного текстового файла (text18-11.txt) вывести на экран его содержимое, количество знаков препинания. Сформировать новый файл, в который поместить строку наименьшей длины.

Вариант 12.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Максимальный элемент:
Среднее арифметическое элементов первой трети:

2. Из предложенного текстового файла (text18-12.txt) вывести на экран его содержимое, количество пробельных символов. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно вставив после каждой строки строку из символов «*».

Вариант 13.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Индекс первого максимального элемента:

Произведение элементов средней трети:

2. Из предложенного текстового файла (text18-13.txt) вывести на экран его содержимое, количество символов в тексте. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно вставив после строки N (N – задается пользователем) произвольную фразу.

Вариант 14.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Индекс первого минимального элемента:
Умножаем все элементы на минимальный элемент:

2. Из предложенного текстового файла (text18-14.txt) вывести на экран его содержимое, количество пробельных символов. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы третей строки их числовыми кодами.

Вариант 15.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Индекс последнего минимального элемента:
Умножаем все элементы на первый элемент:

2. Из предложенного текстового файла (text18-15.txt) вывести на экран его содержимое, количество букв в нижнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы нижнего регистра на верхний.

Вариант 16.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:

Положительные числа:
Количество положительных чисел:
Отрицательные числа:
Количество отрицательных чисел:

2. Из предложенного текстового файла (text18-16.txt) вывести на экран его содержимое, количество букв в верхнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив все знаки пунктуации на знак «!».

Вариант 17.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Произведение элементов:
Количество пар, для которых произведение элементов делится на 3 (элементы пары в последовательности являются соседними):

2. Из предложенного текстового файла (text18-17.txt) вывести на экран его содержимое, количество знаков препинания. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между первой и второй.

Вариант 18.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Среднее арифметическое элементов:
Последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов:

2. Из предложенного текстового файла (text18-18.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках. Сформировать новый файл, в который поместить текст в стихотворной форме выведя строки в обратном порядке.

Вариант 19.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Сумма элементов:
Элементы до n-1 умножены на элемент n:

2. Из предложенного текстового файла (text18-19.txt) вывести на экран его содержимое, количество символов, принадлежащих к группе букв. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы верхнего регистра на нижний.

Вариант 20.
1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:
Количество элементов:
Минимальный элемент:
Числа кратные трем:
Количество чисел кратных трем:

2. Из предложенного текстового файла (text18-20.txt) вывести на экран его содержимое, количество символов в тексте. Сформировать новый файл, в который поместить строку наибольшей длины.

Вариант 21.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Содержимое первого файла:
Отрицательные элементы:
Количество отрицательных элементов:
Среднее арифметическое:

Содержимое второго файла:
Положительные элементы:
Количество положительных элементов:
Сумма положительных элементов:

2. Из предложенного текстового файла (text18-21.txt) вывести на экран его содержимое, количество знаков препинания. Сформировать новый файл, в который поместить текст в стихотворной форме выведя строки в обратном порядке.

Вариант 22.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Содержимое первого файла:
Четные элементы:
Количество четных элементов:
Среднее арифметическое:

Содержимое второго файла:
Нечетные элементы:
Количество нечетных элементов:
Сумма положительных элементов:

2. Из предложенного текстового файла (text18-22.txt) вывести на экран его содержимое, количество букв в верхнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы третей строки их числовыми кодами.

Вариант 23.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Содержимое первого файла:
Четные элементы:
Произведение четных элементов:
Минимальный элемент:

Содержимое второго файла:
Нечетные элементы:
Количество нечетных элементов:
Сумма нечетных элементов:

2. Из предложенного текстового файла (text18-23.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы верхнего регистра на нижний.

Вариант 24.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Содержимое первого файла:
Элементы кратные 3:
Произведение элементов:
Минимальный элемент:

Содержимое второго файла:
Элементы кратные 5:
Количество элементов:
Среднее арифметическое элементов:

2. Из предложенного текстового файла (text18-24.txt) вывести на экран его содержимое, количество символов, принадлежащих к группе букв. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы нижнего регистра на верхний.

Вариант 25.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Элементы после сортировки:
Количество элементов:
Минимальный элемент кратный 2:
Максимальный элемент кратный 5:

2. Из предложенного текстового файла (text18-25.txt) вывести на экран его содержимое, количество символов, принадлежащих к группе букв. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно удалив букву «с» из текста.

Вариант 26.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Элементы первого файла, отсутствующие во втором:
Элементы второго файла, отсутствующие в первом:
Количество элементов:
Индекс первого минимального элемента:
Индекс последнего максимального элемента:

2. Из предложенного текстового файла (text18-26.txt) вывести на экран его содержимое,

количество знаков препинания. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив все знаки пунктуации на знак «/».

Вариант 27.

1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Элементы первого файла, присутствующие во втором:
Элементы второго файла, присутствующие в первом:
Количество элементов:
Количество отрицательных элементов:
Количество положительных элементов:

2. Из предложенного текстового файла (text18-27.txt) вывести на экран его содержимое, количество пробельных символов. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку фразой введенной пользователем.

Вариант 28.

1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Среднее арифметическое элементов первого и второго файлов:
Количество нечетных элементов первого и второго файлов:
Элементы общие для двух файлов:
Количество элементов, общих для двух файлов:

2. Из предложенного текстового файла (text18-28.txt) вывести на экран его содержимое, количество символов в тексте. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно вставив после строки N (N – задается пользователем) произвольную фразу.

Вариант 29.

1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Количество элементов первого и второго файлов:
Количество элементов, общих для двух файлов:

Количество четных элементов первого файла:
Количество нечетных элементов второго файла:

2. Из предложенного текстового файла (text18-29.txt) вывести на экран его содержимое, количество символов в тексте. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между второй и третьей.

Вариант 30.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Количество элементов первого и второго файлов:
Индекс первого минимально элемента первого файла:
Индекс последнего максимального элемента второго файла:
Элементы кратные 4 первого и второго файлов:

2. Из предложенного текстового файла (text18-30.txt) вывести на экран его содержимое, количество знаков препинания. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив после последней строки автора и название произведения.

Вариант 31.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Количество элементов первого и второго файлов:
Элементы первой трети:
Минимальный элемент первой трети:

2. Из предложенного текстового файла (text18-31.txt) вывести на экран его содержимое, количество символов, принадлежащих к группе букв. Сформировать новый файл, в который поместить строку наименьшей длины.

Вариант 32.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:

Количество элементов первого и второго файлов:
Элементы последней трети:
Индекс максимального элемента последней трети:

2. Из предложенного текстового файла (text18-32.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно вставив после каждой строки строку из символов «*».

Вариант 33.
1. Средствами языка Python сформировать два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Элементы первого и второго файлов:
Количество элементов первого и второго файлов:
Сортировка элементов первого и второго файлов:
Элементы кратные 3 второго файла:

2. Из предложенного текстового файла (text18-33.txt) вывести на экран его содержимое, количество пробельных символов. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно вставив после строки N (N – задается пользователем) произвольную фразу.

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Тема 2.1
Принципы модульного программирования

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности

- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части

Ответьте на вопросы.

- Что такое модульное программирование?
- В чём преимущества модульного программирования?
- Как Python ищет модули при импорте?
- Что такое пакет?
- Как импортировать модуль или его содержимое из пакета?
- Как делается перегрузка операторов в Python?
- Какие области видимости существуют в Python?
- Что такое PEP 8?
- Как структурировать пространство имён модуля?
- Какие исключения бывают в Python?

Устный опрос оценивается:
Оценка «5» – полный развернутый ответ;
Оценка «4» – полный неразвернутый ответ;
Оценка «3» – неполный ответ;
Оценка «2» – ответа нет.

Практическое занятие №8

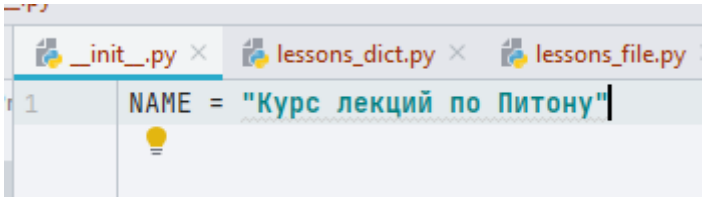
Создание модулей

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с использованием пакетов в IDE PyCharm Community.

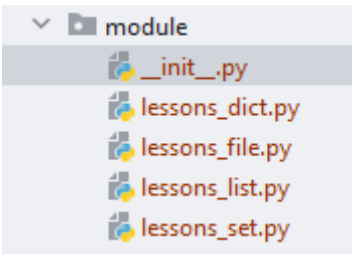
Инструкция к практическому заданию

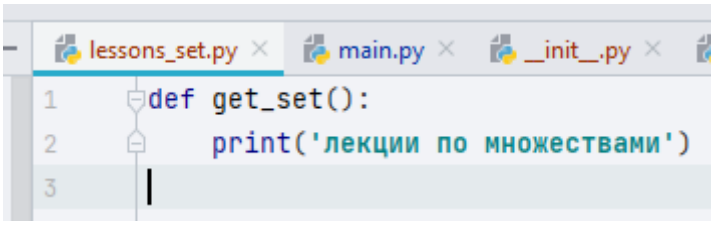
1. Выполнить все указанные действия, в конспекте отразить основные теоретические вопросы.

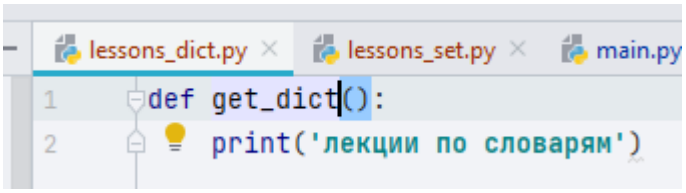
Пакетом называется папка с модулями, как правило, решающими сходные задачи, в которой расположен файл инициализации __init__.py. Файл инициализации может быть пустым или содержать код, который будет выполнен при первой операции импортирования любого модуля, входящего в состав пакета. В любом случае он обязательно должен присутствовать внутри папки с модулями.

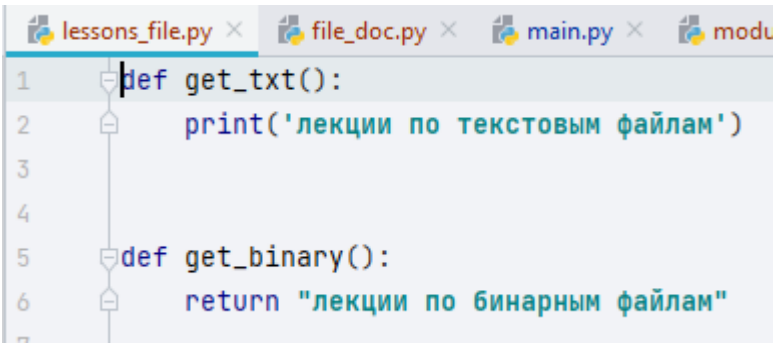


В рамках текущего проекта создадим пакет module





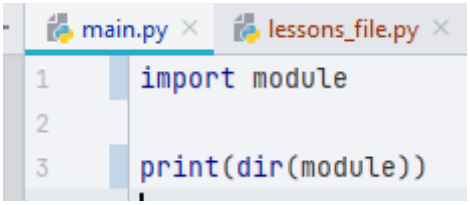




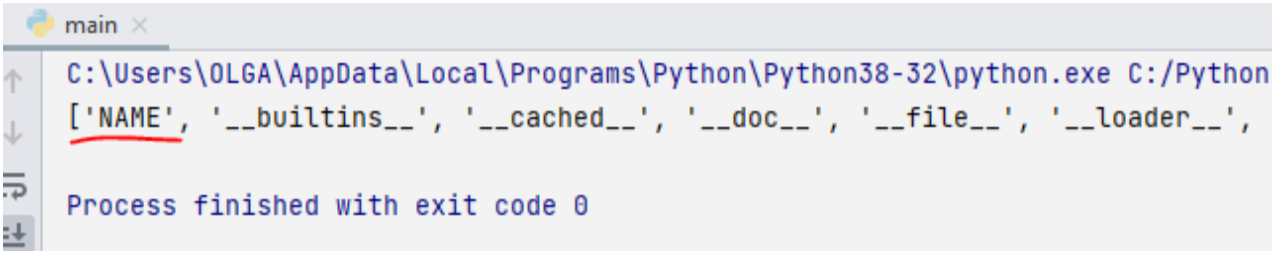
!!! При написании скриптов следить за кодировкой (UTF-8) !!!

В файле main.py выполним импортирование пакета module. Пакеты

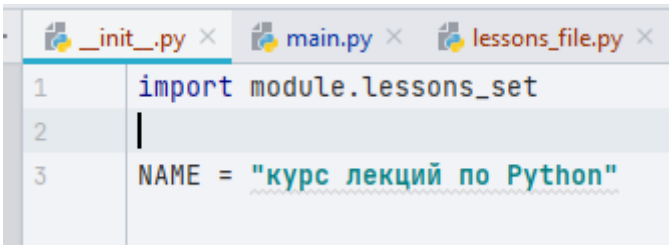
импортируются также, как и модули.



Dir() покажет что импортировалось в пространство имен.

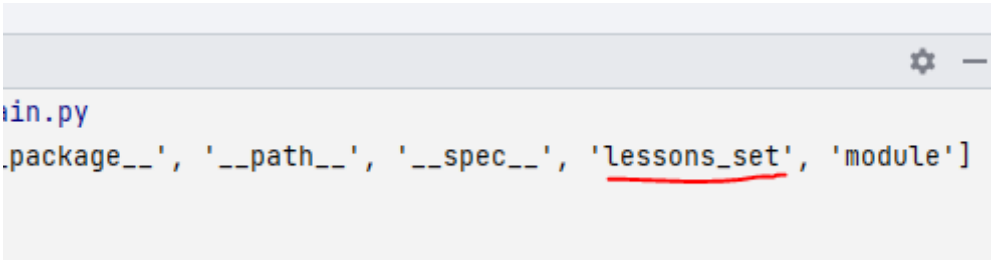


Видим переменную NAME и служебные переменные, т.е. при импорте пакета автоматически выполнится __init__.py и все что в нем определено импортируется при импорте всего пакета. Но пока без других скриптов пакета. Чтобы импортировать скрипт из модуля необходимо

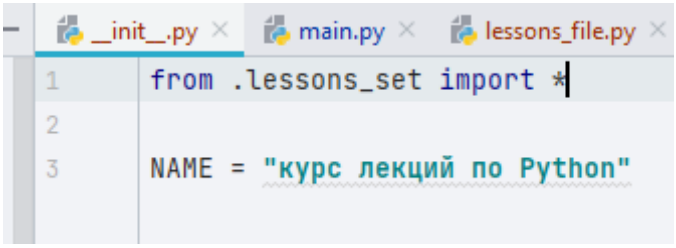


Конструкция module.lessons_set указывает из какого пакета (module) импортируется нужный модуль (lessons_set).

После запуска main.py появилось пространство имен lessons_set, в котором содержатся все объекты скрипта lessons_set, в частности функция get_set().

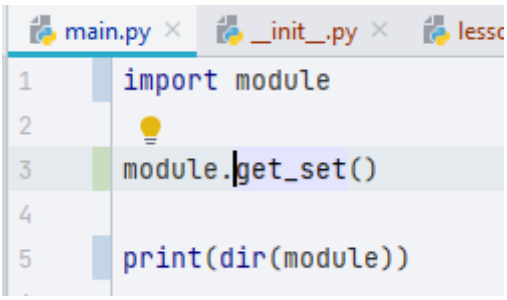


Что бы обратится к функции get_set()необходимо



Конструкция .lessons_set является относительным импортом. Относительный импорт является предпочтительнее, чем абсолютный (module.lessons_set), поскольку имя пакета в будущем может меняться и придется изменять название пакета у каждого import.

Т.к. при использовании import * происходит автоматическое импортирование пространства имен модуля в пространство имен пакета, то в main.py теперь достаточно сразу записать get_set(), минуя lessons_set.

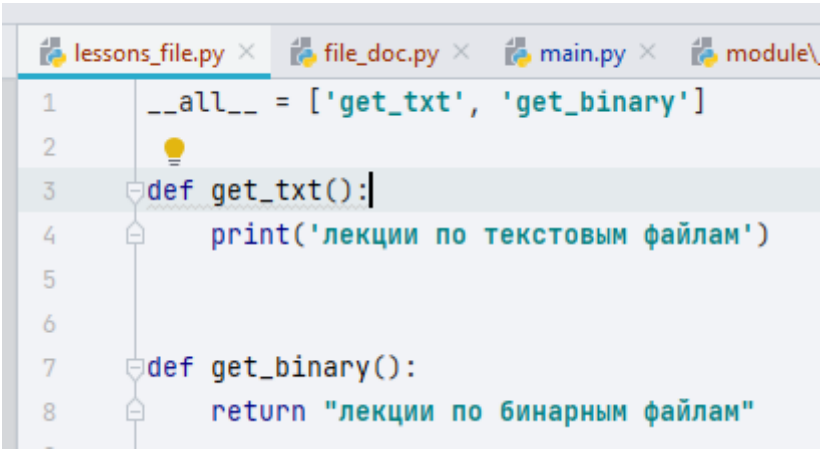


Кроме этого удобно использовать и такой вариант import, при котором все модули импортируются в программу, но имя пакета не указывается:

from . import lessons_set, lessons_dict, lessons_file, lessons_list

Символ «*» не рекомендуется использовать в основных программах (что бы избежать конфликта имен), но для модулей сделано исключение.

Что бы избежать конфликта имен при импортировании (через import *) можно контролировать импортируемые данные с помощью переменной__all__. Переменная __all__ должна ссылаться на список, в котором указываются импортируемые объекты.



В main.py и init.py добавим и выполним

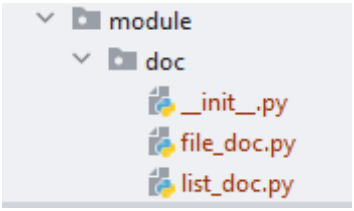
```
main.py | __init__.py | lessons_dict.py | lessons_file.py
1  from module.lessons_set import *
2  from module.lessons_file import *
3
4  NAME = "Курс лекций по Питону"
```

```
main.py | lessons_file.py
1  import module
2
3  module.get_set()
4
5  module.get_txt()
6
7  print(dir(module))
```

```
C:\Users\OLGA\AppData\Local\Prog
лекции по множествами
лекции по текстовым файлам
['NAME', '__builtins__', '__cach
```

Вложенные пакеты

Внутри пакета module создадим пакет doc, в котором тоже будет свой файл инициализации. В пакете doc создадим еще два файла:




```
file_doc.py | doc\__init__.py | module\__init__.py | mai
1  doc = """Раздаточный материал по файлам"""
2
```

В файле init__.py пакета doc выполним импорт двух модулей

```
doc\__init__.py | file_doc.py | module\__init__.py
1  from . import list_doc, file_doc
2
```

В файле init__.py пакета module импортируем пакет doc

```
module\__init__.py   doc\__init__.py   file_
1  from .lessons_set import *
2  from .lessons_file import *
3  from .doc import *
4
5  NAME = "курс лекций по Python"
```

А в файле main.py обратимся к строке «Раздаточный материал по спискам»

```
main.py   doc\__init__.py   module\_
1  import module
2
3  module.get_set()
4
5  module.get_txt()
6
7  print(module.list_doc.doc)
8
9  print(module.file_doc.doc)
10
11 print(module.get_binary())
12
13 print(dir(module))
```

```
Run:  main
C:\Users\OLGA\AppData\Local\Programs\Python
лекции по множествам
лекции по текстовым файлам
Раздаточный материал по спискам
Раздаточный материал по файлам
лекции по бинарным файлам
None
['NAME', '__builtins__', '__cached__', '__d
```

2. Выполнить на оценку.

Создайте пакет 'figures', состоящий из трех подпакетов: 'triangle', 'circle', 'square'. В каждом подпакете будем иметь файл code.py, где создадим ряд функций:

– для пакета 'circle': функции circle_perimeter() – вычисляет длину окружности,

circle_area() – вычисляет площадь окружности.

Еще заведем переменную default_radius = 5, которая будет скрыта при импорте модуля. Ее назначение – дефолтный радиус для окружности, если пользователь не введет свой.

Обе функции принимают на вход только радиус.

– для пакета 'triangle': функции triangle_perimeter() – вычисляет периметр треугольника, triangle_area() – вычисляет площадь фигуры.

Дополнительно создадим три переменные (длины сторон треугольника): a = 7, b = 2, c = 8, которые также не будут видны при импорте.

На вход функциям передается длина трех сторон (если пользователь ничего не введет, то используются значения по умолчанию).

– для пакета 'square': функции square_perimeter() – вычисляет периметр квадрата, square_area() – вычисляет площадь фигуры.

Дополнительная переменная a = 15 не доступна при импорте и принимается функциями, если пользователь не предоставил свои размеры стороны квадрата.

Ваша итоговая задача – позволить человеку, загрузившему ваш пакет, иметь возможность напрямую импортировать все функции из подпакетов.

Например, он может написать так: 'from figures import circle_area'.

Также вы, как разработчик, после написания всей библиотеки решили поменять ее имя на 'figures'.

Постарайтесь сделать код таким, чтобы это не заставило вас переписывать все внутренние импорты с учетом нового именования.

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Тема 3.1
Итераторы и генераторы функциональный стиль программирования

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности

- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части

Ответьте на вопросы.

- Что такое итератор?
- Как использовать итератор?
- Что такое генератор?
- Для чего используется ключевое слово yield?
- В чём разница между генераторами и итераторами?
- Как создать свой итератор?
- Как работать с генераторами?
- Какие преимущества итераторов и генераторов?
- Какие библиотеки используют итераторы и генераторы?
- Какие задачи можно решать с помощью итераторов и генераторов?
- Понятие, применение лямбда функций.
- Назначение функции map().
- Пример map() и лямбда функции (на доске).
- Назначение функции filter().
- Пример filter () и лямбда функции (на доске).
- Назначение функции reduce().
- Пример reduce() и лямбда функции (на доске).

– Преимущества List comprehension
– Пример списковых включений (на доске).
– Пример списковых включений с if (на доске).
– Назначение функции zip().
– Пример работы zip с тремя списками (на доске).
– Понятие и пример итерабельного объекта.
– Понятие и пример итератора.
– Пример выражения-генератора (на доске).
– Отличие функции-генератора от обычной функции. Описание работы.
– Пример функции-генератора (на доске).

Устный опрос оценивается:
Оценка «5» – полный развернутый ответ;
Оценка «4» – полный неразвернутый ответ;
Оценка «3» – неполный ответ;
Оценка «2» – ответа нет.

Практическое занятие №9

Составление программ в функциональном стиле. Составление программ с матрицами

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с использованием списковых включений, итераторов, генераторов, матриц в IDE PyCharm Community.

Инструкция к практическому заданию

Задание 1

Пример решения задачи

```
def prod(lst):
    prod = 1
    for i in lst:
        prod *= i
    return prod

lst = list(map(int, input("введи через элементы списка:  ").split()))
print(prod(lst))
```

Пояснения. Код программы не должен содержать синтаксических и логических ошибок, содержать комментарии, соответствовать PEP 8.

Программы реализовать с использованием списковых включений, итераторов, генераторов.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Варианты заданий

Вариант 1.

1. Организовать и вывести последовательность A из n чисел. Из последовательности A получить две последовательности B и C: в последовательности B – четные элементы A, в C – нечетные элементы A. Произвести суммирование соответствующих элементов последовательностей B и C. Найти минимальный элемент полученной последовательности.
2. Из списка: ['Валентин', 'Петр', 'Анна', 'Евгений', 'Константин', 'Валерия', 'Юлия'] получить новый список, в котором длина слов не превышает 5 символов.

Вариант 2.

1. Организовать и вывести последовательность A из n чисел (n - четное). Из последовательности A получить две последовательности B и C: в последовательности B – первая половина элементов A, в C – вторая половина элементов A. Найти произведение соответствующих элементов последовательностей B и C. Найти среднее арифметической полученной последовательности.
2. Составить список, в который будут включены только согласные буквы и привести их к верхнему регистру. Список: ['Оттава', 'Москва', 'Пекин', 'Полоцк', 'Версаль', 'Дели', 'Каир'].

Вариант 3.

1. Организовать и вывести последовательность из N случайных целых чисел. Из исходной последовательности организовать первую последовательность, содержащую числа кратные трем, и вторую – для всех остальных. Найти количество элементов в полученных последовательностях.
2. Составить генератор (yield), который выводит из строки только цифры.

Вариант 4.

1. В последовательности на n целых чисел умножить элементы до n-1 на элемент n.
2. Составить генератор (yield), который выводит из строки только буквы.

Вариант 5.

1.Из последовательности на n целых чисел создать новую последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов.

2. Составить генератор (yield), который переведет символы строки из нижнего регистра в верхний.

Вариант 6.

1.Даны температуры за месяц март. Необходимо найти количество положительных и отрицательных значений температур в месяце, самую низкую и самую высокую температуры, а также среднемесячное значение температуры.

2. Составить генератор (yield), который переведет символы строки из верхнего регистра в нижний.

Вариант 7.

1.Организовать и вывести последовательность из N случайных целых чисел. Из исходной последовательности организовать первую последовательность, содержащую четные числа, и вторую – для всех остальных. Найти среднее арифметическое в полученных последовательностях.

2. Составить генератор (yield), который преобразует все буквенные символы в строчные.

Вариант 8.

1.В последовательности на n целых элементов найти количество пар, для которых произведение элементов делится на 3 (элементы пары в последовательности являются соседними).

2. Составить генератор (yield), который преобразует все буквенные символы в заглавные.

Вариант 9.

1.Даны две последовательности. Найти элементы, общие для двух последовательностей и их количество.

2. Из заданной строки отобразить только символы нижнего регистра. Использоватьбиблиотеку string. Строка'In PyCharm, you can specify third-party standalone applications and run them as External Tools'.

Вариант 10.

1.В последовательности на n целых чисел найти и вывести:

1. максимальный среди отрицательных
2. элементы кратные двум
3. их сумму

2. Из заданной строки отобразить только цифры. Использоватьбиблиотеку string. Строка - TheGreatPyramidofKhufuatGizawasbuiltabout 2700 BC, 755 feet (230metres) longand 481 feet (147 metres) high.

Вариант 11.

1.В последовательности на n целых чисел найти и вывести:

1. минимальный среди положительных
2. элементы кратные пяти
3. их среднее арифметическое

2.Иззаданнойстрокиотобразить только символы пунктуации. Использовать библиотеку string.

Строка: --msg-template="$FileDir$\{path}:{line}:{column}:{C}:({symbol}){msg}"

Вариант 12.

1.Организовать и вывести последовательность из N случайных целых чисел. Из исходной последовательности организовать новую последовательность, содержащую положительные числа. Найти их количество.

2. Из списка: ['Валентин', 'Петр', 'Анна', 'Евгений', 'Константин', 'Валерия', 'Юлия'] получить новый список, в котором длина слов не превышает 5 символов.

Вариант 13.

1.Проверить есть ли в последовательности целых N чисел число K.

2.Составить список, в который будут включены только согласные буквы и привести их к верхнему регистру. Список: ['Оттава', 'Москва', 'Пекин', 'Полоцк', 'Версаль', 'Дели', 'Каир'].

Вариант 14.

1.Организовать и вывести последовательность из N случайных целых чисел. Из исходной последовательности организовать последовательность, содержащую положительные числа и последовательность, содержащую отрицательные числа. Найти количество элементов в полученных последовательностях.

2. Составить генератор (yield), который выводит из строки только цифры.

Вариант 15.

1.В последовательности на n целых чисел найти и вывести:

1. максимальный среди положительных
2. минимальный среди отрицательных
3. произведение элементов

2.Составить генератор (yield), который выводит из строки только буквы.

Вариант 16.

1.В последовательности на n целых чисел умножить все элементы на первый элемент.

2.Составить генератор (yield), который переведет символы строки из нижнего регистра в верхний.

Вариант 17.

1.В последовательности на n целых чисел умножить все элементы на последний минимальный элемент.

2.Составить генератор (yield), который переведет символы строки из верхнего регистра в нижний.

Вариант 18.

1.В последовательности на n целых элементов найти произведение элементов средней трети.

2.Составить генератор (yield), который преобразует все буквенные символы в строчные.

Вариант 19.

1.В последовательности на n целых элементов найти среднее арифметическое элементов первой трети.

2.Составить генератор (yield), который преобразует все буквенные символы в

заглавные.

Вариант 20.

1.В последовательности на n целых элементов в первой ее половине найти количество положительных элементов.

2.Из заданной строки отобразить только символы нижнего регистра. Использоватьбиблиотеку string. Строка'In PyCharm, you can specify third-party standalone applications and run them as External Tools'.

Вариант 21.

1.Даны две последовательности. Найти элементы, различные для двух последовательностей и их среднее арифметическое.

2.Из заданной строки отобразить только цифры. Использоватьбиблиотекystring. Строка - TheGreatPyramidofKhufuatGizawasbuiltabout 2700 BC, 755 feet (230metres) longand 481 feet (147 metres) high.

Вариант 22.

1.Даны текущие оценки студента по дисциплине «Основы программирования» за месяц. Необходимо найти количество «2», «3», «4» и «5», полученных студентом, и определить итоговую оценку за месяц.

2.Из заданной строки отобразить только символы пунктуации. Использовать библиотеку string.

Строка: --msg-template="$FileDir$\{path}:{line}:{column}:{C}:({symbol}){msg}"

Вариант 23.

1.В последовательности на n целых элементов в последней ее половине найти сумму элементов.

2.Из списка: ['Валентин', 'Петр', 'Анна', 'Евгений', 'Константин', 'Валерия', 'Юлия'] получить новый список, в котором длина слов не превышает 5 символов.

Вариант 24.

1.Даны значения роста 20 юношей. Определить сколько юношей будут направлены в баскетбольную команду (рост от 190) и сколько в футбольную (остальные).

2.Составить список, в который будут включены только согласные буквы и привести их к верхнему регистру. Список: ['Оттава', 'Москва', 'Пекин', 'Полоцк', 'Версаль', 'Дели', 'Каир'].

Вариант 25.

1.Дана последовательность целых чисел. Поменять местами ее первую и последнюю трети.

2.Составить генератор (yield), который выводит из строки только цифры.

Вариант 26.

1.В последовательности их N чисел (N –четное) во второй ее половине найти сумму элементов больших 10.

2.Составить генератор (yield), который выводит из строки только буквы.

Вариант 27.

1.В последовательности их N чисел (N –четное) в первой ее половине найти произведение элементов меньших 0.

2.Составить генератор (yield), который переведет символы строки из нижнего регистра в верхний.

Вариант 28.

1. Организовать и вывести последовательность из 20 целых чисел, выбрать не повторяющиеся элементы, найти их количество. Элементы больше 5 увеличить в два раза.
2. Составить генератор (yield), который переведет символы строки из верхнего регистра в нижний.

Вариант 29.

1. Организовать и вывести последовательность на N произвольных целых элементов, сформировать новую последовательность куда поместить положительные четные элементы, найти их сумму и среднее арифметическое.
2. Составить генератор (yield), который преобразует все буквенные символы в строчные.

Вариант 30.

1.Даны средние значения температур за каждый месяц в году. Найти минимальное и максимальное значения температур за год. Вывести значения температур по временам года.

2.Составить генератор (yield), который преобразует все буквенные символы в заглавные.

Вариант 31.

1. Организовать и вывести последовательность на N произвольных целых элементов, сформировать новую последовательность куда поместить квадраты четных элементов, найти их сумму и среднее арифметическое.
2. Из заданной строки отобразить только символы нижнего регистра. Использоватьбиблиотеку string. Строка 'In PyCharm, you can specify third-party standalone applications and run them as External Tools'.

Вариант 32.

1.В последовательности на n целых чисел умножить все элементы на первый максимальный элемент.

2.Из заданной строки отобразить только символы пунктуации. Использовать библиотеку string.

Строка: --msg-template="$FileDir$\{path}:{line}:{column}:{C}:({symbol}){msg}"

Вариант 33.

1. Организовать и вывести последовательность на N произвольных целых элементов, сформировать новую последовательность куда поместить отрицательные нечетные элементы, найти их сумму и среднее арифметическое.

2. Из заданной строки отобразить только цифры. Использоватьбиблиотеку string. Строка - The Great Pyramid of Khufu at Giza was built about 2700 BC, 755 feet (230metres) long and 481 feet (147 metres) high.

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Задание 2

Пример решения задачи

```
# Организовать и вывести последовательность из N случайных целых чисел.
# Из исходной последовательности организовать
# новую последовательность, содержащую положительные числа.
# Найти их количество.

from random import randint

list1 = [randint(-50, 50) for i in range(15)]
list2 = [i for i in list1 if i > 0]

print('Исходный список: ', list1)
print('Положительные числа: ', list2)
print('Количество положительных чисел: ', len(list2))
```

Пояснения.

Программы реализовать с использованием списковых включений, генераторов.

Код программы не должен содержать синтаксических и логических ошибок, содержать комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Варианты заданий

Вариант 1.

1. Перенести в новую матрицу Matr1 элементы, которые не находятся в первых и последних строках и столбцах матрицы Matr2 произвольного размера.

2. В матрице отрицательные элементы возвести в квадрат.

Вариант 2.

1. В матрице найти минимальный и максимальные элементы.
2. В матрице найти сумму отрицательных элементов в первой трети матрицы.

Вариант 3.

1. В квадратной матрице элементы на главной диагонали увеличить в 2 раза.
2. Из матрицы сформировать массив из положительных четных элементов, найти их сумму и среднее арифметическое.

Вариант 4.

1. В квадратной матрице все элементы, не лежащие на главной диагонали увеличить в 2 раза.
2. Если в матрице имеются положительные элементы, то вывести TRUE, иначе FALSE.

Вариант 5.

1. В матрице элементы второго столбца возвести в квадрат.
2. Сгенерировать матрицу, в которой нечетные элементы заменяются на 0.

Вариант 6.

1. В матрице элементы первого столбца возвести в куб.
2. Сгенерировать матрицу, в которой элементы больше 10 заменяются на 0.

Вариант 7.

1. В матрице элементы строки N (N задать с клавиатуры) увеличить на 3.
2. В матрице элементы последнего столбца заменить на -1.

Вариант 8.

1. В матрице элементы столбца N (N задать с клавиатуры) увеличить в два раза.
2. В матрице элементы последней строки заменить на 0.

Вариант 9.

1. В матрице элементы второго столбца заменить элементами из одномерного динамического массива соответствующей размерности.
2. В матрице найти среднее арифметическое положительных элементов, кратных 3.

Вариант 10.

1. В матрице элементы третьей строки заменить элементами из одномерного динамического массива соответствующей размерности.
2. В матрице найти среднее арифметическое положительных элементов.

Вариант 11.

1. В матрице найти сумму и произведение элементов строки N (N задать с клавиатуры).
2. В матрице найти сумму элементов второй половины матрицы.

Вариант 12.

1. В матрице найти сумму и произведение элементов столбца N (N задать с клавиатуры).
2. В матрице найти отрицательные элементы, сформировать из них новый массив. Вывести размер полученного массива.

Вариант 13.

1. Для каждой строки матрицы с нечетным номером найти среднее арифметическое ее элементов.
2. В матрице найти максимальный положительный элемент, кратный 4.

Вариант 14.

1. Для каждого столбца матрицы с четным номером найти сумму ее элементов.
2. В матрице найти минимальный элемент в предпоследнем столбце.

Вариант 15.

1. В матрице найти суммы элементов каждого столбца и поместить их в новый массив. Выполнить замену элементов второй строки исходной матрицы на полученные суммы.
2. В матрице найти минимальный элемент в предпоследней строке.

Вариант 16.

1. В матрице найти суммы элементов каждой строки и поместить их в новый массив. Выполнить замену элементов третьего столбца исходной матрицы на полученные суммы.
2. В матрице найти сумму элементов второй половины матрицы.

Вариант 17.

1. Сгенерировать матрицу на произвольное количество элементов, в которой задается преобразование от предыдущего элемента к следующему на произвольное значение.
2. В матрице найти сумму элементов первых двух строк.

Вариант 18.

1. В матрице элементы кратные 3 увеличить в 3 раза.

2. В матрице найти среднее арифметическое элементов последних двух столбцов.

Вариант 19.

1. В матрице найти среднее арифметическое элементов последних двух столбцов.
2. Перенести в новую матрицу Matr1 элементы, которые не находятся в первых и последних сроках и столбцах матрицы Matr2 произвольного размера.

Вариант 20.

1. В матрице найти сумму элементов первых двух строк.
2. В матрице найти минимальный и максимальные элементы.

Вариант 21.

1. В матрице найти минимальный элемент в предпоследней строке.
2. В квадратной матрице элементы на главной диагонали увеличить в 2 раза.

Вариант 22.

1. В матрице найти минимальный элемент в предпоследнем столбце.
2. Для каждой строки матрицы с нечетным номером найти среднее арифметическое ее элементов.

Вариант 23.

1. В матрице найти максимальный положительный элемент, кратный 4.
2. В квадратной матрице все элементы, не лежащие на главной диагонали увеличить в 2 раза.

Вариант 24.

1. В матрице найти отрицательные элементы, сформировать из них новый массив. Вывести размер полученного массива.
2. Для каждой строки матрицы с нечетным номером найти среднее арифметическое ее элементов.

Вариант 25.

1. В матрице найти сумму элементов второй половины матрицы.
2. В матрице элементы второго столбца возвести в квадрат.

Вариант 26.

1. В матрице найти среднее арифметическое положительных элементов.
2. В матрице элементы первого столбца возвести в куб.

Вариант 27.

1. В матрице найти среднее арифметическое положительных элементов, кратных 3.
2. В матрице элементы строки N (N задать с клавиатуры) увеличить на 3.

Вариант 28.

1. В матрице элементы последней строки заменить на 0.
2. В матрице элементы столбца N (N задать с клавиатуры) увеличить в два раза.

Вариант 29.

1. В матрице элементы последнего столбца заменить на -1.
2. В матрице элементы третьей строки заменить элементами из одномерного динамического массива соответствующей размерности.

Вариант 30.

1. Сгенерировать матрицу, в которой элементы больше 10 заменяются на 0.
2. В квадратной матрице все элементы, не лежащие на главной диагонали увеличить в 2 раза.

Вариант 31.

1. Сгенерировать матрицу, в которой нечетные элементы заменяются на 0.
2. В матрице элементы второго столбца заменить элементами из одномерного динамического массива соответствующей размерности.

Вариант 32.

1. Если в матрице имеются положительные элементы, то вывести TRUE, иначе FALSE.
2. В матрице найти сумму и произведение элементов строки N (N задать с клавиатуры).

Вариант 33.

1. Из матрицы сформировать массив из положительных четных элементов, найти их сумму и среднее арифметическое.
2. В матрице найти сумму и произведение элементов столбца N (N задать с клавиатуры).

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Тема 4.1
Понятие библиотеки, установка и библиотека OS

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части

Ответьте на вопросы.

- Что такое библиотека в контексте программирования на Python?
- Как установить стороннюю библиотеку в Python с помощью pip?
- В чем разница между библиотекой и модулем в Python?
- Какие шаги необходимо выполнить перед установкой библиотеки в Python?
- Как узнать, какие библиотеки уже установлены в вашей среде Python?
- Какие проблемы могут возникнуть при установке библиотеки и как их решить?
- Как установить определенную версию библиотеки с помощью pip?

- Как обновить уже установленную библиотеку до последней версии?
- Как удалить библиотеку из среды Python?
- Как использовать виртуальные среды для управления зависимостями проекта в Python?
- Назначение библиотек в программировании.
- Порядок работы с библиотекой tkinter.
- Назначение упаковщика.
- На какие действия реагирует обработчик событий?
- Как выглядит обработчик событий tkinter в Питоне?
- Какая функция в библиотеке OS Python используется для получения текущего рабочего каталога?
- Как можно изменить текущий рабочий каталог с помощью библиотеки OS?
- Какой метод из библиотеки OS позволяет создать новый каталог?
- Как можно удалить файл с использованием библиотеки OS?
- Какая функция в библиотеке OS используется для переименования файлов или каталогов?
- Как проверить, существует ли определенный путь, используя библиотеку OS?
- Как можно получить список всех файлов и подкаталогов в указанном каталоге с помощью библиотеки OS?

Устный опрос оценивается:
Оценка «5» – полный развернутый ответ;
Оценка «4» – полный неразвернутый ответ;
Оценка «3» – неполный ответ;
Оценка «2» – ответа нет.

Практическое занятие №10

Составление программ с использованием оконной библиотеки Tkinter. Составление программ с использованием библиотеки OS

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с использованием GUI Tkinter в IDE PyCharm Community.

Инструкция к практическому заданию

Пример решения задачи

```
from tkinter import *

root = Tk()
root.title("Форма регистрации")
root.geometry('505x680+200+200')

img = PhotoImage(file='icon.png')
l=Label(image=img).place(x=1,y=100)

Button(text="Sign Up",width=15,height=2,bg='#00AA72',fg='white',font='arial 18').place(x=17,y=17)
Button(text="Log In",width=15,height=2,bg='grey',fg='white',font='arial 18').place(x=260,y=17)

Label(text="Sign Up for Free",width=25,fg='#0000aa',font='arial 25').place(x=17,y=150)

Label(text="First Name",width=8,fg='#0000aa',font='arial 8').place(x=17,y=220)
Label(text="Last Name",width=8,fg='#0000aa',font='arial 8').place(x=260,y=220)
Entry(width=10, font='arial 28').place(x=17,y=250)
Label(text="Email Adress",width=12,fg='#0000aa',font='arial 8').place(x=17,y=303)
Entry(width=10, font='arial 28').place(x=260,y=250)
Entry(width=22, font='arial 28').place(x=17,y=330)
Label(text="Set A Password",width=12,fg='#0000aa',font='arial 8').place(x=17,y=385)
Entry(width=22, font='arial 28').place(x=17,y=405)

Button(text="GET STARTED",width=19,height=1,bg='#00AA72',fg='white',font='arial 30').place(x=17,y=500)

root.mainloop()
```

Пояснения. Код программы не должен содержать синтаксических и логических ошибок, содержать комментарии, соответствовать PEP 8. Разместить на GitHub проект по практическому занятию.

Варианты заданий

Задание 1. В соответствии с номером варианта перейти по ссылке на прототип. Реализовать его в IDE PyCharm Community с применением пакета tk. Получить интерфейс максимально приближенный к оригиналу (см. таблицу 1). Задание 2. Разработать программу с применением пакета tk, взяв в качестве условия одну любую задачу из ПЗ №№ 1 – 9.

Вариант 1	https://freshdesignweb.com/wp-content/uploads/site/Elegant-Register-Forms-Template.jpg
Вариант 2	https://professorweb.ru/my/html/html5/level2/files/img46023.jpg
Вариант 3	https://cf.ppt-online.org/files/slide/v/vRAlxeaYzgFDVC2krBdPXQtZbW5McoLJ08In6q/slide-21.jpg
Вариант 4	https://lh5.googleusercontent.com/-wG_YHAIbVZU/Ud696wJg0FI/AAAAAAAACP4/eaIzPTZRixE/w59

	6-h642-no/4_3.png
Вариант 5	https://uicookies.com/wp-content/uploads/2019/05/Reg-Form-v5.jpg
Вариант 6	https://i.pinimg.com/originals/73/c6/0d/73c60def8c55043f9fd27b370530a9cf.jpg
Вариант 7	https://www.formget.com/wp-content/uploads/2014/06/style1.png
Вариант 8	https://i.stack.imgur.com/5MatP.png
Вариант 9	https://www.zoho.com/creator/images/subpages/htmlforms/workshop_registration.gif
Вариант 10	https://bramus.github.io/ws1-sws-course-materials/assets/03/testform.jpg
Вариант 11	https://www.bestfree.ru/uslugi/constructors/SozdanieSaytaNaUcoz_3.png
Вариант 12	https://lh5.googleusercontent.com/-wG_YHAIbVZU/Ud696wJg0FI/AAAAAAAACP4/eaIzPTZRixE/w596-h642-no/4_3.png
Вариант 13	https://www.digiseller.ru/preview/125077/p1_30116215520413.JPG
Вариант 14	https://i.pinimg.com/originals/73/c6/0d/73c60def8c55043f9fd27b370530a9cf.jpg
Вариант 15	https://files.codegrape.com/62259/screenshot2.png
Вариант 16	https://studfile.net/html/2706/360/html_uTKkMTCo1E.hFYH/htmlconvd-efYkCR62x1.jpg
Вариант 17	https://www.zoho.com/creator/images/subpages/htmlforms/meeting_registration.gif

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Тема 4.2
Библиотека SQlite

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части

Ответьте на вопросы

- Что такое SQLite?
- Перечислите список стандартных команд SQLite?
- Что такое транзакции SQLite?
- Перечислите преимущества SQLite?
- Как в SQLite хранятся булевы значения?
- Назовите команду, которая используется для создания базы данных в SQLite?
- Как отобразить все столбцы в базе данных SQLite с помощью Python?

– Как изменить таблицу SQLite с помощью Python?

.

Практическое занятие №11

Составление программ для работы с базами данных

Цели закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, работы с БД в IDE PyCharm Community.

Пример решения задачи

```
import sqlite3 as sq
from data_users import info_users

with sq.connect('saper.db') as con:
    cur = con.cursor()
    # cur.execute("DROP TABLE IF EXISTS users")
    cur.execute("""CREATE TABLE IF NOT EXISTS users (
        user_id INTEGER PRIMARY KEY AUTOINCREMENT,
        name TEXT NOT NULL,
        sex INTEGER NOT NULL DEFAULT 1,
        old INTEGER,
        score INTEGER
    )""")

with sq.connect('saper.db') as con:
    cur = con.cursor()
    cur.execute("INSERT INTO users VALUES (1, 'olga', 2, 22, 1000)")

with sq.connect('saper.db') as con:
    cur = con.cursor()
    cur.executemany("INSERT INTO users VALUES (?, ?, ?, ?, ?)", info_users)

with sq.connect('saper.db') as con:
    cur = con.cursor()
    cur.execute("SELECT * FROM users WHERE score > 900 ORDER BY score DESC")
    result = cur.fetchall()
    print(result)
```

Пояснения.

Средствами Python реализовать программу для работы с однотабличной БД. Программа должна обеспечивать функционал по вводу данных в БД (10 позиций), их поиску, удалению и

редактированию. При организации поиска, удаления и редактирования использовать WHERE, предусмотреть по три SQL-запроса для каждой операции. Структура БД указана в каждом варианте.

Структуру БД и расширение функционала можно изменять по согласованию с преподавателем.

Код программы не должен содержать синтаксических и логических ошибок, содержать обработку исключений и комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы.

Разместить на GitHub проект и отчет по практическому занятию.

Варианты заданий

Вариант 1

Приложение ***ОПТОВАЯ БАЗА*** для автоматизированного контроля движения товаров на оптовой базе. Таблица ***Товары*** должна содержать следующие данные: *Код товара, Наименование товара, Наименование магазина, Заявки магазина, Количество товара на складе, Единицы измерения, Оптовая цена.*

Вариант 2

Приложение ***АБИТУРИЕНТ*** для автоматизации работы приемной комиссии, которая обеспечивает обработку анкетных данных абитуриентов. Таблица ***Анкета*** содержит следующие данные об абитуриентах: *Регистрационный номер, Фамилия, Имя, Отчество, Дата Рождения, Награды* (наличие кр. Диплома или медали (да/нет)), *Адрес,* выбранная *Специальность.*

Вариант 3

Приложение ***КАФЕДРА*** для автоматизации работы отдела кадров ВУЗа. Таблица ***Преподавательский состав*** должна содержать следующие данные: *Табельный номер, Фамилия И.О.*, *Дата рождения, Должность, Ученая степень, Нагрузка, Зарплата.*

Вариант 4

Приложение ***БИБЛИОТЕКА*** для автоматизированного контроля литературных источников в библиотеке. БД должна содержать таблицу ***Каталог*** с информацией о книгах и следующей структурой записи: *Код книги, Жанр, Страна издания, Серия, Автор, Название книги, Год выпуска, Аннотация*.

Вариант 5

Приложение ***БАНК*** для отслеживания накапливаемых на счетах клиентов банка сумм. Таблица ***Клиент*** должна содержать следующую информацию: *Код клиента, Клиент* (Ф.И.О.), *Периодический платеж, Годовой %, Срок вклада, Пластиковая карта* (логическое поле), *Конечная сумма*.

Вариант 6

Приложение для туристического агентства ***ТУР***. Таблица ***Турист*** должна содержать следующую информацию о клиентах турфирмы: *Код клиента, Клиент* (Фамилия), *Телефон*, *Название страны*, *Регион*, *Продолжительность поездки*, *Стоимость путевки*.

Вариант 7

Приложение ***АПТЕКА*** для автоматизации работы аптечных пунктов. Таблица ***Лекарственные Средства*** должна содержать следующую информацию: *Код, Название препарата, Применение*, *Количество, Цена, Страна-производитель.*

Вариант 8

Приложение ***ЗАКАЗЫ ТОВАРОВ*** для автоматизированного контроля заказов торговой фирмы. Таблица ***Заказы*** должна содержать информацию о товарах со следующей структурой записи: *Код товара, Наименование товара, Заказчик* (наименование организации, возможны повторяющиеся значения), *Дата заказа, Срок исполнения* (от 1 до 10 дней), *Стоимость заказа*.

Вариант 9

Приложение ***УЧЕБНЫЙ ПЛАН*** для автоматизированного контроля учебной

нагрузки по кафедре. Таблица ***Дисциплины*** должна иметь следующую структуру записи: *Код дисциплины, Наименование дисциплины, Специальность, Лекции* (колич.часов)*, Практические* (колич.часов)*, Лабораторные* (колич.часов)*, Форма отчетности*.

Вариант 10

Приложение ***ПРОМЫШЛЕННОСТЬ*** для автоматизированного учета информации о промышленных предприятиях республики. БД содержит таблицу ***Предприятия***, имеющую следующую структуру записи: *Код предприятия, Наименование предприятия, Физический адрес, Филиалы* (количество филиалов), *Общая числ. персонала, Общая стоим. оборудования, объем выпускаемой продукции, Дата регистрации.*

Вариант 11

Приложение ***КОММАНДИРОВОЧНЫЕ РАСХОДЫ*** для автоматизированного финансового контроля на предприятии. БД должна содержать таблицу ***Статьи расходов***, имеющую следующую структуру записи: *№ приказа*, *Фамилия*, Место *командировки*, *Оплата*, *Аванс*, *Вид расходов*, *Сумма расходов*.

Вариант 12

Приложение ***ТЕЛЕМАСТЕРСКАЯ*** для автоматизированного контроля работ по ремонту бытовой техники. БД должна содержать таблицу ***Ремонт телевизоров***, имеющую следующую структуру записи: *Марка телевизора*, *Завод-изготовитель*, *Цена*, *Дата ремонта*, *Документ*, *Мастер*, *Сумма оплаты*.

Вариант 13

Приложение ***ТОВАРНЫЙ ЗАПАС*** для автоматизированного учета товарных запасов на складе. БД должна содержать таблицу ***Товары*** со следующей структурой записи: *Код товара*, *Торговая марка, Тип, Цена*, *Количество на складе*, *Минимальный запас.*

Вариант 14

Приложение ***ТОРГОВАЯ ФИРМА*** для автоматизированного контроля продаж товаров торговой фирмы. БД должна содержать таблицу ***Продажа товаров*** со следующей структурой записи: *Дата продажи*, *Товар, Сумма, Скидка*, *Филиал, Менеджер.*

Вариант 15

Приложение ***РАСХОДЫ ПО ВИДАМ ПРОДУКЦИИ*** для автоматизированного контроля затрат на производство продукции. БД должна содержать таблицу ***Расходы*** со следующей структурой записи: *Дата, Код продукта, Наименование продукта,*

Расходы, *Сумма.*

Вариант 16

Приложение ***КОНТРОЛЬ ИСПОЛНЕНИЯ ПОРУЧЕНИЙ*** для некоторой организации. БД должна содержать таблицу ***Поручения*** соследующей структурой записи: Порядковый номер поручения, Название поручения, Дата выдачи поручения, Срок исполнения, Исполнитель.

Вариант 17

Приложение ***БЮРО ПО ТРУДОУСТРОЙСТВУ*** для некоторой организации. БД должна содержать таблицу ***Работник*** со следующей структурой записи: фамилия, имя, отчество, возраст, пол, профессия, стаж работы.

Вариант 18

Приложение ***СТРАХОВАЯ КОМПАНИЯ*** для некоторой организации. БД должна содержать таблицу ***Договор*** со следующей структурой записи: дата заключения, страховая сумма, вид страхования, тарифная ставка и филиал, в котором заключался договор.

Вариант 19

Приложение ***ГРУЗОВЫЕ ПЕРЕВОЗКИ*** для некоторой организации. БД должна содержать таблицу ***Перевозки*** со следующей структурой записи: маршрут, фамилия водителя, даты отправки и прибытия, масса груза.

Вариант 20

Приложение ***НОТАРИАЛЬНАЯ КОНТОРА*** для некоторой организации. БД должна содержать таблицу ***Нотариальные услуги*** со следующей структурой записи: ФИО клиента, услуга, сумма сделки, комиссионные (доход конторы).

Вариант 21

Приложение ***РАСПРЕДЕЛЕНИЕ ДОПОЛНИТЕЛЬНЫХ ОБЯЗАННОСТЕЙ*** для некоторой организации. БД должна содержать таблицу ***Обязанности*** со следующей структурой записи: ФИО работника, вид дополнительной работы, сумма оплаты, срок.

Вариант 22

Приложение ***УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ*** для некоторой организации. БД должна содержать таблицу ***Канцелярия*** со следующей структурой записи: ФИО работника, отдел, вид расхода, сумма.

Вариант 23

Приложение ***ПРОКАТ АВТОМОБИЛЕЙ*** для некоторой организации. БД должна содержать таблицу ***Клиент*** со следующей структурой записи: ФИО клиента, марка авто, срок проката, сумма, предоплата (да/нет).

Вариант 24

Приложение ***ВЫДАЧА КРЕДИТОВ*** для некоторой организации. БД должна содержать таблицу ***Клиент*** со следующей структурой записи: ФИО клиента, ФИО сотрудника банка, срок кредита, процент кредита, сумма кредита.

Вариант 25

Приложение ***ПЛАТНАЯ ПОЛИКЛИНИКА*** для некоторой организации. БД должна содержать таблицу ***Пациент*** со следующей структурой записи: ФИО пациента, ФИО врача, диагноз, стоимость лечение.

Вариант 26

Приложение ***ЮВЕЛИРНАЯ МАСТЕРСКАЯ*** для некоторой организации. БД должна содержать таблицу ***Изделие*** со следующей структурой записи: ФИО клиента, ФИО мастера, вид изделия, материал, стоимость работ.

Вариант 27

Приложение ***ИНТЕРНЕТ-МАГАЗИН*** для некоторой организации. БД должна содержать таблицу ***Продажи*** со следующей структурой записи: ФИО покупателя, товар, единицу измерения (штуки, килограммы, литры), количество, стоимость.

Вариант 28

Приложение ***ПАРИКМАХЕРСКАЯ*** для некоторой организации. БД должна содержать таблицу ***Услуги*** со следующей структурой записи: ФИО мастера, ФИО клиента, пол, название стрижки, стоимость.

Вариант 29

Приложение ***ХИМЧИСТКА*** для некоторой организации. БД должна содержать таблицу ***Услуги*** со следующей структурой записи: ФИО мастера, ФИО клиента, тип чистки, стоимость, скидка.

Вариант 30

Приложение ***СДАЧА В АРЕНДУ ТОРГОВЫХ ПЛОЩАДЕЙ*** для некоторой организации. БД должна содержать таблицу ***Торговая точка*** со следующей структурой записи: этаж, площадь, наличие кондиционера и стоимость аренды в день.

Вариант 31

Приложение ***СДАЧА В АРЕНДУ ТОРГОВЫХ ПЛОЩАДЕЙ*** для некоторой организации. БД должна содержать таблицу ***Клиент*** со следующей структурой записи: ФИО клиента, код помещения, срок аренды, стоимость аренды за весь срок.

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – Практическое занятие не выполнено.

Тема 5.1
Базовые понятия ООП

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде
- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Текст задания по теоретической части

Ответьте на вопросы.

- Понятие ООП;
- Основные концепции ООП;
- Понятие класса; чем являются переменные и функции в классе;
- Понятие наследования;
- Понятие подкласса;
- Понятие суперкласса;
- Поведение подкласса к суперклассу.
- Понятие, назначение конструктора класса.
- Что значит перегрузка метода?

Устный опрос оценивается:
Оценка «5» – полный развернутый ответ;
Оценка «4» – полный неразвернутый ответ;
Оценка «3» – неполный ответ;
Оценка «2» – ответа нет.

Практическое занятие №12

Составление программ различной тематики (ООП). Составление программ с использованием библиотеки pickle

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с ООП, приобрести навыки составления программ с использованием библиотеки pickle.

Пример решения задачи

```
class User:
    def __init__(self, age, name):
        self.age = age
        self.name = name

    def get_values(self):
        print(f"{self.age = }")
        print(f"{self.name = }")

import pickle

with open("User.bin", "wb") as file:
    pickle.dump(User, file)

with open("User.bin", "rb") as file:
   user = pickle.load(file)

a = User(10, "Max")
a.get_values()
```

Пояснения.

Из каждого блока заданий выполнить одну задачу из своего варианта (суммарно 2 задачи).

Код программы не должен содержать синтаксических и логических ошибок, содержать комментарии, соответствовать PEP 8.

Отчет должен содержать постановку задачи, текст программы на Python, протокол работы программы. Код программы должен содержать все тестовые запуски (для всех реализуемых классов).

Варианты заданий

Блок заданий 1

1. Создайте класс «Круг», который имеет атрибут радиуса и методы для вычисления площади, длины окружности и диаметра.
2. Создайте класс «Студент», который имеет атрибуты имя, фамилия и оценки. Добавьте методы для вычисления среднего балла и определения, является ли студент отличником.

3. Создайте класс «Счетчик», который имеет атрибут текущего значения и методы для инкремента и декремента значения.
4. Создайте класс «Календарь», который имеет атрибуты год, месяц и день. Добавьте методы для определения дня недели, проверки на високосный год и определения количества дней в месяце.
5. Создайте класс «Банк», который имеет атрибуты суммы денег и процентной ставки. Добавьте методы для вычисления процентных начислений и снятия денег.
6. Создайте класс «Книга», который имеет атрибуты название, автор и количество страниц. Добавьте методы для чтения и записи книги.
7. Создайте класс «Матрица», который имеет атрибуты количества строк и столбцов. Добавьте методы для сложения, вычитания и умножения матриц.
8. Создайте класс "Человек" с атрибутами "имя", "возраст" и "пол". Напишите метод, который выводит информацию о человеке в формате "Имя: имя, Возраст: возраст, Пол: пол".
9. Создайте класс "Калькулятор" с методами "сложение", "вычитание", "умножение" и "деление". Каждый метод должен принимать два аргумента и возвращать результат операции.
10. Создайте класс "Здание" с атрибутами "адрес" и "количество этажей". Напишите метод, который выводит информацию о здании в формате "Адрес: адрес, Количество этажей: этажи".
11. Создайте класс "Товар" с атрибутами "название", "цена" и "количество". Напишите метод, который выводит информацию о товаре в формате "Название: название, Цена: цена, Количество: кол-во".
12. Создайте класс "Животное" с атрибутами "имя" и "вид". Напишите метод, который выводит информацию о животном в формате "Имя: имя, Вид: вид".
13. Создайте класс "Компьютер" с атрибутами "марка", "процессор" и "оперативная память". Напишите метод, который выводит информацию о компьютере в формате "Марка: марка, Процессор: процессор, Оперативная память: память".
14. Создайте класс "Машина" с атрибутами "марка", "модель" и "год выпуска". Напишите метод, который выводит информацию о машине в формате "Марка: марка, Модель: модель, Год выпуска: год".
15. Создайте класс «Банк», который имеет атрибуты суммы денег и процентной ставки. Добавьте методы для вычисления процентных начислений и снятия денег.
16. Создайте класс «Книга», который имеет атрибуты название, автор и количество страниц. Добавьте методы для чтения и записи книги.
17. Создайте класс «Матрица», который имеет атрибуты количества строк и столбцов. Добавьте методы для сложения, вычитания и умножения матриц.
18. Создайте класс «Круг», который имеет атрибут радиуса и методы для вычисления площади, длины окружности и диаметра.
19. Создайте класс «Студент», который имеет атрибуты имя, фамилия и оценки. Добавьте методы для вычисления среднего балла и определения, является ли студент отличником.
20. Создайте класс «Счетчик», который имеет атрибут текущего значения и методы для инкремента и декремента значения.
21. Создайте класс «Календарь», который имеет атрибуты год, месяц и день. Добавьте методы для определения дня недели, проверки на високосный год и определения количества дней в месяце.

22. Создайте класс "Товар" с атрибутами "название", "цена" и "количество". Напишите метод, который выводит информацию о товаре в формате "Название: название, Цена: цена, Количество: кол-во".
23. Создайте класс "Животное" с атрибутами "имя" и "вид". Напишите метод, который выводит информацию о животном в формате "Имя: имя, Вид: вид".
24. Создайте класс "Компьютер" с атрибутами "марка", "процессор" и "оперативная память". Напишите метод, который выводит информацию о компьютере в формате "Марка: марка, Процессор: процессор, Оперативная память: память".
25. Создайте класс "Машина" с атрибутами "марка", "модель" и "год выпуска". Напишите метод, который выводит информацию о машине в формате "Марка: марка, Модель: модель, Год выпуска: год".
26. Создайте класс «Круг», который имеет атрибут радиуса и методы для вычисления площади, длины окружности и диаметра.
27. Создайте класс «Студент», который имеет атрибуты имя, фамилия и оценки. Добавьте методы для вычисления среднего балла и определения, является ли студент отличником.
28. Создайте класс «Счетчик», который имеет атрибут текущего значения и методы для инкремента и декремента значения.
29. Создайте класс «Календарь», который имеет атрибуты год, месяц и день. Добавьте методы для определения дня недели, проверки на високосный год и определения количества дней в месяце.
30. Создайте класс «Банк», который имеет атрибуты суммы денег и процентной ставки. Добавьте методы для вычисления процентных начислений и снятия денег.
31. Создайте класс «Книга», который имеет атрибуты название, автор и количество страниц. Добавьте методы для чтения и записи книги.
32. Создайте класс «Матрица», который имеет атрибуты количества строк и столбцов. Добавьте методы для сложения, вычитания и умножения матриц.
33. Создайте класс "Человек" с атрибутами "имя", "возраст" и "пол". Напишите метод, который выводит информацию о человеке в формате "Имя: имя, Возраст: возраст, Пол: пол".

Блок заданий 2

1. Создайте класс "Фигура", который содержит метод расчета площади фигуры. Создайте классы "Квадрат" и "Прямоугольник", которые наследуются от класса "Фигура". Каждый класс должен иметь метод расчета площади собственной фигуры.
2. Создайте класс "Человек", который содержит информацию о имени, возрасте и поле. Создайте классы "Мужчина" и "Женщина", которые наследуются от класса "Человек". Каждый класс должен иметь метод, который выводит информацию о поле объекта.
3. Создайте класс "Автомобиль", который содержит информацию о марке, модели и годе выпуска. Создайте класс "Грузовик", который наследуется от класса "Автомобиль" и содержит информацию о грузоподъемности. Создайте класс "Легковой автомобиль", который наследуется от класса "Автомобиль" и содержит информацию о количестве пассажиров.

4. Создайте класс "Животное", который содержит информацию о виде и возрасте животного. Создайте классы "Собака" и "Кошка", которые наследуются от класса "Животное" и содержат информацию о породе.
5. Создайте класс "Фрукт", который содержит информацию о наименовании и весе фрукта. Создайте классы "Яблоко" и "Апельсин", которые наследуются от класса "Фрукт" и содержат информацию о цвете.
6. Создание базового класса "Фигура" и его наследование для создания классов "Квадрат", "Прямоугольник" и "Круг". Класс "Фигура" будет иметь общие методы, такие как вычисление площади и периметра, а классы-наследники будут иметь специфичные методы и свойства.
7. Создание базового класса "Транспортное средство" и его наследование для создания классов "Автомобиль" и "Мотоцикл". В классе "Транспортное средство" будут общие свойства, такие как максимальная скорость и количество колес, а классы-наследники будут иметь свои уникальные свойства и методы.
8. Создание базового класса "Животное" и его наследование для создания классов "Собака" и "Кошка". В классе "Животное" будут общие методы, такие как "дышать" и "питаться", а классы-наследники будут иметь свои уникальные методы и свойства, такие как "гавкать" и "мурлыкать".
9. Создание базового класса "Работник" и его наследование для создания классов "Менеджер" и "Инженер". В классе "Работник" будут общие методы, такие как "работать" и "получать зарплату", а классы-наследники будут иметь свои уникальные методы и свойства, такие как "управлять командой" и "проектировать системы".
10. Создайте базовый класс "Животное" со свойствами "вид", "количество лап", "цвет шерсти". От этого класса унаследуйте класс "Собака" и добавьте в него свойства "кличка" и "порода".
11. Создайте базовый класс "Фигура" со свойствами "ширина" и "высота". От этого класса унаследуйте классы "Прямоугольник" и "Квадрат". Для класса "Квадрат" переопределите методы, связанные с вычислением площади и периметра.
12. Создайте базовый класс "Транспорт" со свойствами "марка", "модель" и "год выпуска". От этого класса унаследуйте класс "Автомобиль" и добавьте в него свойство "тип кузова".
13. Создайте базовый класс "Человек" со свойствами "имя", "возраст" и "пол". От этого класса унаследуйте классы "Мужчина" и "Женщина" и добавьте в них свойства, связанные с социальным положением (например, "семейное положение", "количество детей" и т.д.).
14. Создайте базовый класс "Форма" со свойствами "цвет" и "тип". От этого класса унаследуйте класс "Круг" и добавьте в него свойство "радиус". Определите методы вычисления площади и периметра.
15. Создайте класс "Животное", который содержит информацию о виде и возрасте животного. Создайте классы "Собака" и "Кошка", которые наследуются от класса "Животное" и содержат информацию о породе.
16. Создайте класс "Фрукт", который содержит информацию о наименовании и весе фрукта. Создайте классы "Яблоко" и "Апельсин", которые наследуются от класса "Фрукт" и содержат информацию о цвете.
17. Создание базового класса "Фигура" и его наследование для создания классов "Квадрат", "Прямоугольник" и "Круг". Класс "Фигура" будет иметь общие

методы, такие как вычисление площади и периметра, а классы-наследники будут иметь специфичные методы и свойства.

18. Создание базового класса "Транспортное средство" и его наследование для создания классов "Автомобиль" и "Мотоцикл". В классе "Транспортное средство" будут общие свойства, такие как максимальная скорость и количество колес, а классы-наследники будут иметь свои уникальные свойства и методы.
19. Создание базового класса "Животное" и его наследование для создания классов "Собака" и "Кошка". В классе "Животное" будут общие методы, такие как "дышать" и "питаться", а классы-наследники будут иметь свои уникальные методы и свойства, такие как "гавкать" и "мурлыкать".
20. Создание базового класса "Работник" и его наследование для создания классов "Менеджер" и "Инженер". В классе "Работник" будут общие методы, такие как "работать" и "получать зарплату", а классы-наследники будут иметь свои уникальные методы и свойства, такие как "управлять командой" и "проектировать системы".
21. Создайте базовый класс "Животное" со свойствами "вид", "количество лап", "цвет шерсти". От этого класса унаследуйте класс "Собака" и добавьте в него свойства "кличка" и "порода".
22. Создайте базовый класс "Фигура" со свойствами "ширина" и "высота". От этого класса унаследуйте классы "Прямоугольник" и "Квадрат". Для класса "Квадрат" переопределите методы, связанные с вычислением площади и периметра.
23. Создайте класс "Фигура", который содержит метод расчета площади фигуры. Создайте классы "Квадрат" и "Прямоугольник", которые наследуются от класса "Фигура". Каждый класс должен иметь метод расчета площади собственной фигуры.
24. Создайте класс "Человек", который содержит информацию о имени, возрасте и поле. Создайте классы "Мужчина" и "Женщина", которые наследуются от класса "Человек". Каждый класс должен иметь метод, который выводит информацию о поле объекта.
25. Создайте класс "Автомобиль", который содержит информацию о марке, модели и годе выпуска. Создайте класс "Грузовик", который наследуется от класса "Автомобиль" и содержит информацию о грузоподъемности. Создайте класс "Легковой автомобиль", который наследуется от класса "Автомобиль" и содержит информацию о количестве пассажиров.
26. Создайте класс "Человек", который содержит информацию о имени, возрасте и поле. Создайте классы "Мужчина" и "Женщина", которые наследуются от класса "Человек". Каждый класс должен иметь метод, который выводит информацию о поле объекта.
27. Создайте класс "Автомобиль", который содержит информацию о марке, модели и годе выпуска. Создайте класс "Грузовик", который наследуется от класса "Автомобиль" и содержит информацию о грузоподъемности. Создайте класс "Легковой автомобиль", который наследуется от класса "Автомобиль" и содержит информацию о количестве пассажиров.
28. Создайте класс "Животное", который содержит информацию о виде и возрасте животного. Создайте классы "Собака" и "Кошка", которые наследуются от класса "Животное" и содержат информацию о породе.

29. Создайте базовый класс "Животное" со свойствами "вид", "количество лап", "цвет шерсти". От этого класса унаследуйте класс "Собака" и добавьте в него свойства "кличка" и "порода".
30. Создайте базовый класс "Фигура" со свойствами "ширина" и "высота". От этого класса унаследуйте классы "Прямоугольник" и "Квадрат". Для класса "Квадрат" переопределите методы, связанные с вычислением площади и периметра.
31. Создайте базовый класс "Транспорт" со свойствами "марка", "модель" и "год выпуска". От этого класса унаследуйте класс "Автомобиль" и добавьте в него свойство "тип кузова".
32. Создайте базовый класс "Человек" со свойствами "имя", "возраст" и "пол". От этого класса унаследуйте классы "Мужчина" и "Женщина" и добавьте в них свойства, связанные с социальным положением (например, "семейное положение", "количество детей" и т.д.).
33. Создайте класс "Фигура", который содержит метод расчета площади фигуры. Создайте классы "Квадрат" и "Прямоугольник", которые наследуются от класса "Фигура". Каждый класс должен иметь метод расчета площади собственной фигуры.

Блок заданий 3

Для задачи из блока 1 создать две функции, save_def и load_def, которые позволяют сохранять информацию из экземпляров класса (3 шт.) в файл и загружать ее обратно. Использовать модуль pickle для сериализации и десериализации объектов Python в бинарном формате.

Критерии оценивания:

Оценка «5» - решены три задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, в соответствии с условием и пояснениями.

Оценка «4» - решены две задачи, коды программ и отчет размещены на GitHub, задачи решены полностью, с незначительными отклонениями от условий и пояснений.

Оценка «3» - решена одна задача, код программы и отчет предоставлены, задача решена полностью, со значительными отклонениями от условий и пояснений.

Оценка «2» – практическое занятие не выполнено.

Тема 5.2

Визуальное событийно-управляемое программирование

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
- ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
- ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности
- ОК.04 Эффективно взаимодействовать и работать в коллективе и команде

- ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации
- ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение
- ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях
- ОК.08 Использовать средства физической культуры для поддержания здоровья
- ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Практическое занятие №13

Создание проекта с использованием компонентов для работы с текстом. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с ООП, приобрести навыки составления программ с использованием библиотеки pickle.

Пояснения

Разбор используемых компонентов

1. Работа с текстом

- **Entry** (`entry_name`) — однострочное текстовое поле для ввода коротких строк (имя, фамилия). Метод `.get()` забирает введенный текст.
- **Text** (`text_note`) — многострочное текстовое поле для объемных заметок. Метод `.get("1.0", tk.END)` считывает текст с первой строки до конца. [1]

2. Работа с числами

- Используется стандартный виджет **Entry** (`entry_age`), но введенное значение принудительно преобразуется из строки в число с помощью встроенной функции Python `int(...)` или `float(...)`.

3. Работа с датами и временем

- Используется встроенный модуль **`datetime`**:
  - `datetime.strptime(str, "%d.%m.%Y")` — преобразует введенную пользователем строковую дату в полноценный объект даты.
  - `datetime.now()` — получает текущую системную дату и время.

- Разница между объектами дат (`current_date - birth_date`) позволяет вычислить количество прожитых дней.
- `.strftime(...)` форматирует дату обратно в удобный для отображения вид.

4. Отображение результатов

- **Label** (`label_result`) — текстовая метка, которая динамически обновляет свой текст через метод `.config(text=...)` после нажатия на кнопку **Button**.
- **messagebox** — всплывающее диалоговое окно для вывода системных ошибок, если пользователь введет некорректный формат чисел или дат.

Задание

Разработать проект на языке Python с использованием графического интерфейса (**Tkinter**) для работы с текстом, числами, датами и временем — это **«Дневник событий / Калькулятор возраста и дней до даты»**.

```python
from datetime import datetime
import tkinter as tk
from tkinter import messagebox


def calculate_and_save():
  try:
    # 1. Работа с текстом (Entry и Text)
    name = entry_name.get()
    note_text = text_note.get("1.0", tk.END).strip()

    # 2. Работа с числами (Entry -> int/float)
    age = int(entry_age.get())

    # 3. Работа с датой и временем (парсинг строки в дату)
    # Формат ввода даты: ДД.ММ.ГГГГ (например, 15.05.2005)
    birth_date_str = entry_date.get()
    birth_date = datetime.strptime(birth_date_str, "%d.%m.%Y")

    current_date = datetime.now()
    days_diff = (current_date - birth_date).days

    # 4. Вывод результата в текстовое поле отображения (Label / Text)
    result_str = (
        f"Имя: {name}\n"
        f"Возраст: {age} лет\n"
        f"Дата рождения: {birth_date.strftime('%d.%m.%Y')}\n"
        f"Прошло дней с рождения: {days_diff}\n"
        f"Заметка: {note_text}"
    )

    label_result.config(text=result_str)
```

```
  except ValueError:
    messagebox.showerror(
        "Ошибка",
        "Проверьте правильность заполнения полей!\nДата должна быть в формате ДД.ММ
    )


# Создание главного окна
root = tk.Tk()
root.title("Анализатор данных и текста")
root.geometry("400x500")

# Компоненты для ввода ТЕКСТА
tk.Label(root, text="Введите имя:").pack(pady=5)
entry_name = tk.Entry(root, width=30)
entry_name.pack()

tk.Label(root, text="Заметка / Описание:").pack(pady=5)
text_note = tk.Text(root, width=35, height=4)
text_note.pack()

# Компоненты для ввода ЧИСЕЛ
tk.Label(root, text="Введите возраст (число):").pack(pady=5)
entry_age = tk.Entry(root, width=30)
entry_age.pack()

# Компоненты для ввода ДАТЫ
tk.Label(root, text="Дата рождения (ДД.ММ.ГГГГ):").pack(pady=5)
entry_date = tk.Entry(root, width=30)
entry_date.pack()
```

```
# Кнопка запуска действия
btn = tk.Button(
    root, text="Обработать и отобразить", command=calculate_and_save
)
btn.pack(pady=15)

# Компонент для отображения результата
label_result = tk.Label(
    root, text="Здесь появится результат...", justify=tk.LEFT, relief="sunken", wi
)
label_result.pack(pady=10)

root.mainloop()
```

Критерии оценивания:

Оценка «5» - разработанный проект и отчет размещены на GitHub в соответствии с условием и пояснениями.

Оценка «4» разработанный проект и отчет размещены на GitHub с незначительными отклонениями от условий и пояснений.

Оценка «3» - разработанный проект и отчет размещены на GitHub со значительными отклонениями от условий и пояснений.

Оценка «2» – практическое занятие не выполнено.

Практическое занятие №14

События компонентов (элементов управления), их сущность и назначениею

Создание процедур на основе событий

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с ООП.

Пояснения

Событие компонента — это сигнал от графического интерфейса (GUI) о том, что произошло определенное действие, требующее реакции программы. Сущность событий заключается в реализации **событийно-ориентированного**

программирования, где выполнение кода управляется действиями пользователя (клик, ввод текста) или системы (таймер, загрузка).

Назначение событий

- **Интерактивность:** Связывают графические элементы (кнопки, поля ввода) с логикой программы.
- **Асинхронность:** Программа не ждет ввода пользователя в бесконечном цикле, а «спит» и просыпается только в момент совершения действия.
- **Разделение интерфейса и логики:** Позволяют изолировать внешний вид приложения от его функционала.

Задание

В Python для создания GUI часто используется встроенная библиотека **Tkinter**. Чтобы связать событие с функцией (процедурой-обработчиком), используют метод `.bind()` или специальный параметр `command` у компонентов.

Разработать проект согласно коду:

```
import tkinter as tk
from tkinter import messagebox

# 1. Создание процедур-обработчиков (функций, которые выполняются при событии)

def on_button_click():
    """Обработчик для простого нажатия на кнопку (через параметр command)"""
    user_text = entry.get()
    if user_text.strip():
        messagebox.showinfo("Приветствие", f"Привет, {user_text}!")
    else:
        messagebox.showwarning("Предупреждение", "Поле ввода пустое!")

def on_clear_enter(event):
    """Обработчик события наведения мыши (через метод bind).
    Обязательно принимает аргумент event, содержащий данные о событии."""
    clear_button.config(bg="lightcoral")

def on_clear_leave(event):
    """Обработчик события ухода мыши с элемента"""
    clear_button.config(bg="SystemButtonFace") # Возврат стандартного цвета

def clear_text(event):
    """Обработчик двойного клика для очистки поля"""
    entry.delete(0, tk.END)

# 2. Создание главного окна приложения
root = tk.Tk()
root.title("Пример обработки событий")
root.geometry("400x200")
```

```
# 3. Создание компонентов (виджетов)
label = tk.Label(root, text="Введите ваше имя:")
label.pack(pady=5)

entry = tk.Entry(root, width=30)
entry.pack(pady=5)

# Способ 1: Использование параметра command (для простых кликов на кнопки)
submit_button = tk.Button(root, text="Отправить", command=on_button_click)
submit_button.pack(pady=5)

clear_button = tk.Button(root, text="Очистить (Двойной клик)")
clear_button.pack(pady=5)

# Способ 2: Использование метода .bind() для специфических событий
# <Double-Button-1> — двойной щелчок левой кнопкой мыши
clear_button.bind("<Double-Button-1>", clear_text)

# <Enter> — курсор мыши зашел в пределы компонента
clear_button.bind("<Enter>", on_clear_enter)

# <Leave> — курсор мыши покинул пределы компонента
clear_button.bind("<Leave>", on_clear_leave)

# Запуск главного цикла прослушивания событий
root.mainloop()
```

Как это работает:

1. **Параметр command:** Используется у кнопок. Мы просто передаем имя функции on_button_click (без скобок), и Tkinter сам вызывает её при клике.
2. **Метод .bind(шаблон, функция):** Позволяет отслеживать низкоуровневые действия. Передаваемая функция обязательно должна принимать один аргумент (обычно именуемый event), в который передаются координаты мыши, код нажатой клавиши и другие системные данные.

Критерии оценивания:

Оценка «5» - разработанный проект и отчет размещены на GitHub в соответствии с условием и пояснениями.

Оценка «4» разработанный проект и отчет размещены на GitHub с незначительными отклонениями от условий и пояснений.

Оценка «3» - разработанный проект и отчет размещены на GitHub со значительными отклонениями от условий и пояснений.

Оценка «2» – практическое занятие не выполнено.

Тема 5.3
Разработка оконного приложения

Проверяемые результаты обучения по общим и профессиональным компетенциям

- ОК.01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

− ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности

− ОК.03 Планировать и реализовывать профессиональное и личностное развитие, использовать знания правовой и финансовой грамотности

− ОК.04 Эффективно взаимодействовать и работать в коллективе и команде

− ОК.05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации

− ОК.06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение

− ОК.07 Содействовать сохранению окружающей среды, эффективно действовать в чрезвычайных ситуациях

− ОК.08 Использовать средства физической культуры для поддержания здоровья

− ОК.09 Пользоваться профессиональной документацией на государственном и иностранном языках

− ПК 1.8

- использовать методы и приемы формализации поставленных задач

- использовать методы и приемы алгоритмизации поставленных задач

- использовать программное обеспечение для графического отображения алгоритмов

- применять алгоритмы решения типовых задач в соответствующих областях
- осуществлять коммуникации с заинтересованными сторонами

Практическое занятие №15
Создание проекта с использованием кнопочных компонентов, компонентов стандартных диалогов и системы меню

Цели: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ с ООП.

Пояснения

Разбор ключевых компонентов проекта

1. Система меню (tk.Menu)

Меню конструируется иерархически.

- Создается главное меню main_menu, которое привязывается к окну через root.config(menu=...).
- Выпадающие списки (например, Файл, Справка) создаются как отдельные объекты tk.Menu(self.main_menu, tearoff=0).
- Элементы меню добавляются с помощью .add_command(label="Название", command=функция). При нажатии на пункт меню вызывается соответствующий метод.

2. Кнопочные компоненты (ttk.Button)

В проекте используется модуль tkinter.ttk, предоставляющий современные нативные стили для кнопок.

- Кнопки располагаются на специальной панели toolbar (компонент ttk.Frame).
- Свойство command привязывает кнопку к логике приложения. Кнопки «Открыть» и «Сохранить» дублируют функции из главного меню, что является хорошим тоном в UX/UI дизайне.

3. Компоненты стандартных диалогов

Для вызова диалогов используются встроенные подмодули filedialog и messagebox:

- **filedialog.askopenfilename()** — открывает стандартное окно операционной системы для выбора файла. Возвращает путь к файлу или пустую строку, если пользователь нажал «Отмена».
- **filedialog.asksaveasfilename()** — диалог для указания пути и имени сохраняемого файла.
- **messagebox.showinfo()** — всплывающее окно с кнопкой «ОК» для вывода уведомлений.
- **messagebox.askyesno()** — диалоговое окно с вопросом и кнопками «Да» / «Нет». Возвращает True или False, что позволяет делать ветвление логики (например, перед удалением данных).

Задание

Разработать графическое приложение на Python с использованием библиотеки Tkinter. Создать текстовый редактор, в котором будут задействованы **кнопочные компоненты**, **системное меню** и **стандартные диалоговые окна** (выбор файла, предупреждения).

Скопируйте этот код в файл `app.py` и запустите его.

```python
import tkinter as tk
from tkinter import filedialog, messagebox, ttk


class SimpleEditor:

    def __init__(self, root):
        self.root = root
        self.root.title("Мини-редактор текста")
        self.root.geometry("600x400")

        # Настройка главного окна
        self.setup_menu()
        self.setup_toolbar()
        self.setup_text_area()
```

```
    def setup_menu(self):
        """Создание системы меню"""
        self.main_menu = tk.Menu(self.root)
        self.root.config(menu=self.main_menu)

        # Меню "Файл"
        self.file_menu = tk.Menu(self.main_menu, tearoff=0)
        self.main_menu.add_cascade(label="Файл", menu=self.file_menu)
        self.file_menu.add_command(label="Открыть", command=self.open_file)
        self.file_menu.add_command(label="Сохранить", command=self.save_file)
        self.file_menu.add_separator()
        self.file_menu.add_command(label="Выход", command=self.root.quit)

        # Меню "Справка"
        self.help_menu = tk.Menu(self.main_menu, tearoff=0)
        self.main_menu.add_cascade(label="Справка", menu=self.help_menu)
        self.help_menu.add_command(
            label="О программе", command=self.show_about
        )

    def setup_toolbar(self):
        """Создание панели с кнопочными компонентами"""
        self.toolbar = ttk.Frame(self.root, padding="3")
        self.toolbar.pack(side=tk.TOP, fill=tk.X)

        # Кнопки со стилизацией ttk
        self.btn_open = ttk.Button(
            self.toolbar, text="Открыть", command=self.open_file
        )
        self.btn_open.pack(side=tk.LEFT, padx=2)

        self.btn_save = ttk.Button(
            self.toolbar, text="Сохранить", command=self.save_file
        )
        self.btn_save.pack(side=tk.LEFT, padx=2)
```

```
        self.btn_save = ttk.Button(
            self.toolbar, text="Сохранить", command=self.save_file
        )
        self.btn_save.pack(side=tk.LEFT, padx=2)

        self.btn_clear = ttk.Button(
            self.toolbar, text="Очистить", command=self.clear_text
        )
        self.btn_clear.pack(side=tk.LEFT, padx=2)

    def setup_text_area(self):
        """Создание поля для ввода текста"""
        self.text_area = tk.Text(self.root, wrap=tk.WORD)
        self.text_area.pack(fill=tk.BOTH, expand=True, padx=5, pady=5)

    # --- Действия и Стандартные диалоги ---

    def open_file(self):
        """Стандартный диалог открытия файла"""
        file_path = filedialog.askopenfilename(
            title="Выберите текстовый файл",
            filetypes=[("Текстовые файлы", "*.txt"), ("Все файлы", "*.*")],
        )
        if file_path:
            with open(file_path, "r", encoding="utf-8") as file:
                self.text_area.delete("1.0", tk.END)
                self.text_area.insert("1.0", file.read())

    def save_file(self):
        """Стандартный диалог сохранения файла"""
        file_path = filedialog.asksaveasfilename(
            defaultextension=".txt",
            filetypes=[("Текстовые файлы", "*.txt"), ("Все файлы", "*.*")],
        )
```

```
        if file_path:
            with open(file_path, "w", encoding="utf-8") as file:
                file.write(self.text_area.get("1.0", tk.END))
            messagebox.showinfo(
                "Успех", "Файл успешно сохранен!"
            )  # Диалог информации

    def clear_text(self):
        """Диалоговое окно подтверждения перед очисткой"""
        answer = messagebox.askyesno(
            "Подтверждение", "Вы уверены, что хотите очистить весь текст?"
        )
        if answer:
            self.text_area.delete("1.0", tk.END)

    def show_about(self):
        """Стандартный диалог типа 'О программе'"""
        messagebox.showinfo(
            "О программе",
            "Мини-редактор текста\nВерсия 1.0\nСоздано на Python + Tkinter",
        )


if __name__ == "__main__":
    root = tk.Tk()
    app = SimpleEditor(root)
    root.mainloop()
```

Критерии оценивания:

Оценка «5» - разработанный проект и отчет размещены на GitHub в соответствии с условием и пояснениями.

Оценка «4» разработанный проект и отчет размещены на GitHub с незначительными отклонениями от условий и пояснений.

Оценка «3» - разработанный проект и отчет размещены на GitHub со значительными отклонениями от условий и пояснений.

Оценка «2» – практическое занятие не выполнено.

4. Контрольно-оценочные материалы для промежуточной аттестации по учебной дисциплине – ОП.10 «Основы алгоритмизации и программирования на Python».

Перечень заданий и вопросов на зачет:

Теоретические вопросы

1. Сформулируйте основную теорему программирования. Какие три базовые алгоритмические структуры она выделяет?
2. Дайте определение и пример алгоритмической структуры «последовательность».
3. Дайте определение и пример алгоритмической структуры «ветвление».
4. Дайте определение и пример алгоритмической структуры «цикл».
5. Опишите синтаксис и принцип работы условного оператора if-elif-else в Python.
6. Опишите синтаксис и принцип работы циклических операторов while и for.
7. Назовите и объясните назначение операторов управления циклом (break, continue в цикле).
8. Что входит в понятие «составляющие языка программирования»?
9. Раскройте термины «алфавит», «синтаксис», «семантика» применительно к Python.
10. Перечислите правила составления (именования) переменных в Python. Какие имена недопустимы?
11. Назовите основные встроенные типы данных в Python. Какие из них изменяемые, а какие неизменяемые?
12. Опишите работу оператора print(). Для чего используются параметры sep и end?
13. Опишите работу оператора input(). Какой тип данных он возвращает и как выполнить преобразование типов?
14. Что такое функция в Python? Как объявить функцию, вызвать её и вернуть значение?
15. Объясните разницу между локальными и глобальными переменными?
16. Что такое фактические параметры (аргументы) функции?
17. Дайте определение кортежу. Какие свойства (особенности) у кортежа по сравнению со списком?

18. Дайте определение списку. Как создать список, обратиться к элементу по индексу, получить срез?
19. Чем отличается массив от обычного списка в Python?
20. Дайте определение словарю. По какому принципу он работает?
21. Объясните работу метода .append() для списков?
22. Объясните работу метода .split() для строк. Как разбить строку на список и как собрать список обратно в строку?
23. Что такое лямбда-функция? В чем её отличие от обычной функции, объявленной через def? Приведите пример использования.
24. Что такое «динамическая типизация» в Python? Как работает функция type()?
25. Как реализуется множественное ветвление с помощью elif? Приведите синтаксис.
26. В чем разница между циклом с предусловием и циклом со счетчиком? Когда какой выбирать?
27. Что произойдет, если в цикле while условие всегда истинно? Как принудительно прервать выполнение программы?
28. В чем разница между операторами break и continue?
29. Что такое «пустая инструкция» (pass) в Python? Где и зачем она применяется?
30. Что такое «чувствительность к регистру» в синтаксисе Python? Приведите пример.
31. Что такое «семантическая ошибка»? Чем она отличается от синтаксической ошибки?
32. Почему нельзя использовать зарезервированные слова (ключевые слова) в качестве имен переменных? Приведите примеры таких слов.
33. Какие имена переменных считаются «невалидными» (недопустимыми) в Python? Назовите не менее трех случаев.
34. Как преобразовать строку "123" в целое число, а строку "3.14" — в вещественное? Что будет при попытке преобразовать "abc"?
35. Как с помощью print() вывести несколько значений без пробела между ними?
36. Как с помощью print() вывести значения в файл, а не в консоль?
37. Как считать с клавиатуры несколько чисел, введенных в одну строку через пробел, и сохранить их в список?

38. Что такое «функция без возврата значения»? Что фактически возвращает такая функция?
39. Что произойдет, если внутри функции попытаться изменить глобальную переменную без объявления global? Объясните поведение интерпретатора.
40. Что такое «аргументы по умолчанию»? Приведите пример.
41. В чем разница между операциями append() и insert() для списков?
42. Как с помощью метода split() разбить строку на слова? Как ограничить количество разбиений?
43. Как с помощью метода join() объединить список строк в одну строку через запятую?
44. Что произойдет при попытке обратиться к несуществующему ключу словаря? Как безопасно получить значение (метод get())?
45. Может ли ключом словаря быть список? Почему?
46. Приведите пример использования лямбда-функции вместе с map() или filter().
47. Как отсортировать список словарей по значению определенного ключа с помощью sorted() и lambda?
48. В чем разница между методами .remove() и .pop() для списков?

Практические задания

1. **Стоимость конфет:** Известно, что X кг конфет стоит A рублей. Определить, сколько стоит 1 кг и Y кг этих же конфет.
2. **Движение лодки:** Скорость лодки в стоячей воде V км/ч, скорость течения реки U км/ч (U < V). Время движения лодки по озеру T1 ч, а по реке (против течения) — T2 ч. Определить путь S, пройденный лодкой.
3. **Перестановка цифр:** Дано трехзначное число. Вывести число, полученное при перестановке цифр десятков и единиц исходного числа (например, 123 → 132).
4. **Время с начала суток:** С начала суток прошло N секунд (N — целое). Найти количество полных часов, прошедших с начала суток, и количество секунд, прошедших с начала последнего часа.
5. **Четность и знак:** Дано целое число. Если оно является положительным, то прибавить к нему 1; если отрицательным, то вычесть из него 2; если нулевым, то заменить его на 10. Вывести полученное число.

6. **Квадрант координат:** Даны координаты точки, не лежащей на координатных осях OX и OY. Определить номер координатной четверти, в которой находится данная точка.
7. **Високосный год:** Дан номер года (положительное целое число). Определить количество дней в этом году (365 или 366), учитывая, что год високосный, если он делится на 4, но не делится на 100, или делится на 400.
8. **Сравнение с нулем:** Даны три целых числа: A, B, C. Проверить истинность высказывания: «Ровно одно из чисел A, B, C положительное».
9. **Сумма в диапазоне:** Даны два целых числа A и B (A < B). Найти сумму всех целых чисел от A до B включительно (не используя формулу арифметической прогрессии).
10. **Степени двойки:** Дано целое число N (>0). Найти наименьшее целое положительное число K, квадрат которого превосходит N: $K^2 > N$. (Функцию sqrt не использовать).
11. **Ряд 1/N:** Дано число A (>1). Вывести наименьшее из целых чисел K, для которых сумма 1 + 1/2 + ... + 1/K будет больше A, и саму эту сумму.
12. **Обратный порядок цифр:** Дано целое число N (>0). Используя операции деления нацело и взятия остатка от деления, найти число, полученное при прочтении числа N справа налево (реверс числа).
13. **Соседи в списке:** Дан список размера N. Найти номера тех элементов списка, которые больше своего правого соседа, и количество таких элементов.
14. **Сдвиг элементов:** Дан список размера N и целое число K (1 < K < N). Осуществить сдвиг элементов списка вправо на K позиций. Первые K элементов полученного списка положить равными 0.
15. **Локальные минимумы:** Дан список размера N. Найти максимальный из его локальных минимумов (элемент, который меньше любого из своих соседей).
16. **Обмен половин:** Дан список размера N (N — четное число). Поменять местами первый элемент со вторым, третий — с четвертым и т.д.
17. **Списки и условия:** Дан список A размера N. Сформировать новый список B того же размера, где Bk = 2*Ak, если Ak < 5, иначе Bk = Ak/2.
18. **Регистр букв:** Дана строка. Преобразовать в ней все прописные латинские буквы в строчные.
19. **Удаление пробелов:** Дана строка-предложение с избыточными пробелами между словами. Преобразовать ее так, чтобы между словами был ровно один пробел.

20. **Шифровка строки:** Дана строка-предложение. Зашифровать ее, поместив вначале все символы, расположенные на четных позициях, а затем, в обратном порядке, все символы на нечетных позициях.

21. **Подсчет цифр:** Дана строка. Подсчитать количество содержащихся в ней цифр.

22. **Извлечение имени файла:** Дана строка, содержащая полное имя файла (диск, папки, имя, расширение). Выделить из этой строки имя файла (без расширения).

23. **Степени числа:** Описать функцию PowerA234(A, B, C, D), вычисляющую вторую, третью и четвертую степень числа A и возвращающую их в B, C, D. Найти степени 5 чисел.

24. **Периметр и площадь:** Описать функцию TrianglePS(a, P, S), вычисляющую по стороне a равностороннего треугольника его периметр P = 3*a и площадь S = $a^2\sqrt{3}/4$. Найти P и S для трех треугольников.

25. **Цифры числа:** Описать функцию DigitCountSum(K, C, S), находящую количество C цифр числа K и их сумму S. Найти кол-во и сумму цифр для 5 чисел.

26. **Циклический сдвиг:** Описать функцию ShiftRight3(A, B, C), выполняющую правый циклический сдвиг: A → B, B → C, C → A. Выполнить сдвиг для двух наборов чисел.

27. **Фильтрация списка:** Из списка имен ['Валентин', 'Петр', 'Анна', 'Евгений', 'Константин', 'Валерия', 'Юлия'] получить новый список, в котором длина слов не превышает 5 символов.

28. **Генератор чисел:** Организовать и вывести последовательность из N случайных целых чисел. Из исходной последовательности получить новую, содержащую четные числа. Найти их сумму.

29. **Работа со строками:** Из заданной строки отобразить только символы нижнего регистра (используя string или isupper()/islower()).

30. **Списковые включения:** Сгенерировать матрицу (список списков) 5x5, в которой нечетные элементы заменяются на 0.

31. **Обработка чисел из файла:** Дан текстовый файл с целыми числами. Сформировать новый файл, содержащий: исходные данные, количество элементов, максимальный элемент.

32. **Анализ стихотворения:** Из предложенного текстового файла вывести на экран его содержимое и количество знаков препинания. Сформировать новый файл, где строки идут в обратном порядке.

33. **Работа с двумя файлами:** Сформировать два текстовых файла с числами. Создать третий файл, содержащий элементы, общие для двух первых файлов, и их количество.

34. **Температуры:** Даны температуры за месяц март. Найти количество положительных и отрицательных значений, самую низкую и самую высокую температуры, а также среднемесячное значение.

35. **Поиск в матрице:** Дана матрица (двумерный список). Найти сумму элементов второй половины матрицы (условно разделив её по вертикали или горизонтали).

Критерии оценивания:

Оценка «5» - дан ответ на все вопросы в полном объеме и задание решено полностью код не должен содержать синтаксических и логических ошибок, содержать комментарии, соответствовать PEP 8.

Оценка «4» - дан ответ на оба вопроса в неполном объеме / задание решено с допущением логических ошибок /не соответствует PEP 8.

Оценка «3» - дан ответ на все вопросы задание не решено/ не дан ответ на вопросы, задание решено полностью, дан ответ на один вопрос, задание решено с допущением логических ошибок /не соответствует PEP 8.

Оценка «2» – не дан устный ответ, задание не выполнено.

Лист согласования

Дополнения и изменения к комплекту ФОС на учебный год

Дополнения и изменения к комплекту ФОС на _____ учебный год по дисциплине _____ комплект ФОС внесены следующие изменения:

_____

_____

_____

_____

_____

Дополнения и изменения в комплекте ФОС обсуждены на заседании ЦК _____ Протокол № _____ от «_____» _____ 20_____г.

Председатель ЦК