

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

УТВЕРЖДАЮ
Директор колледжа

С.Н.Козлов
29.08.2025

**ИНСТРУМЕНТАЛЬНОЕ ПРОГРАММНОЕ
ОБЕСПЕЧЕНИЕ**

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
ПО ИЗУЧЕНИЮ УЧЕБНОГО ПРЕДМЕТА,
ЗАДАНИЯ НА ДОМАШНЮЮ КОНТРОЛЬНУЮ РАБОТУ
ДЛЯ УЧАЩИХСЯ ЗАОЧНОЙ ФОРМЫ ПОЛУЧЕНИЯ ОБРАЗОВАНИЯ
ПО СПЕЦИАЛЬНОСТИ 5-04-0612-02 «РАЗРАБОТКА И
СОПРОВОЖДЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ИНФОРМАЦИОННЫХ СИСТЕМ»

Автор: Денисовец Д.А., преподаватель высшей квалификационной категории учреждения образования «Могилевский государственный политехнический колледж»

Рецензент: Карманов А.В., преподаватель первой квалификационной категории учреждения образования «Могилевский государственный политехнический колледж»

Разработано на основе учебной программы учреждения образования по учебному предмету «Инструментальное программное обеспечение» профессионального компонента учебного плана учреждения образования по специальности 5-04-0612-02 «Разработка и сопровождение программного обеспечения информационных систем» для реализации образовательной программы среднего специального образования, обеспечивающей получение квалификации специалиста со средним специальным образованием, утвержденной директором колледжа, 2025.

Обсуждено и одобрено
на заседании цикловой комиссии
специальностей в области программного
обеспечения информационных систем
Протокол № 13 от 29.08.2025
Председатель цикловой комиссии
_____Д.А.Денисовец

Пояснительная записка

Учебный предмет «Инструментальное программное обеспечение» предназначен для реализации образовательной программы среднего специального образования по специальности 5-04-0612-02 «Разработка и сопровождение программного обеспечения информационных систем» и предусматривает изучение синтаксиса Python, типов данных, строк, списков, кортежей, словарей, множеств, условных операторов, операторов циклов, функций, обработки исключений, модулей, пакетов, работы с файлами, основ и базовых принципов объектно-ориентированного программирования (далее – ООП), взаимодействия с сетью Интернет, клиент-серверных приложений, сервер-сокетных соединений, фреймворков и библиотек.

Основной целью изучения учебного предмета «Инструментальное программное обеспечение» является формирование профессиональной компетентности будущих специалистов в области разработки программных средств, использования языка программирования при реализации конкретных задач, формирование базовых понятий структурного программирования, ознакомление с объектно-ориентированным языком программирования Python, с возможностями, синтаксисом языка, технологией и методами программирования в среде Python, обучение практическим навыкам программирования на языке Python для решения типовых задач математики и информатики, а также при разработке простейших игр, развитие логики обучающихся.

В процессе преподавания учебного предмета учитываются межпредметные связи программного учебного материала с такими учебными предметами учебного плана учреждения образования по специальности, как «Основы алгоритмизации и программирования», «Конструирование программ и языки программирования», «Системы управления базами данных».

В ходе изложения программного учебного материала необходимо руководствоваться нормативными правовыми актами, техническими нормативными правовыми актами, соблюдать единство терминологии и обозначений, обеспечивать формирование универсальных, профессиональных компетенций, установленных в образовательном стандарте по соответствующей специальности.

В результате изучения учебного предмета «Инструментальное программное обеспечение» учащиеся должны:

знать на уровне представления:
базовый синтаксис языка Python;
основы функционального программирования и ООП в контексте применения их в языке программирования Python;
знать на уровне понимания:
области применения языка Python;
недостатки языка Python;
возможности интеграции языка Python с иными языками программирования;
уметь:
разрабатывать эффективные программы в стиле ООП на языке программирования Python;
пользоваться различными парадигмами программирования в зависимости от задачи;
пользоваться широким спектром библиотек (сборка, установка, настройка, тестирование) и разрабатывать собственные библиотеки;
разрабатывать приложения и веб-приложения с внешними источниками данных (текстовыми файлами, базами данных, xml-файлами).

Учебной программой определены цели изучения каждой темы, спрогнозированы результаты их достижения в соответствии с уровнями усвоения учебного материала.

Для закрепления теоретического материала и формирования у учащихся необходимых умений учебной программой предусмотрено проведение лабораторных работ.

В учебной программе по учебному предмету приведены критерии оценки результатов учебной деятельности учащихся, разработанные в соответствии с Правилами проведения аттестации учащихся, курсантов при освоении содержания образовательных программ среднего специального образования; перечень средств обучения, необходимых для обеспечения образовательного процесса.

В результате изучения учебного предмета учащиеся выполняют 1 домашнюю контрольную работу и 1 обязательную контрольную работу.

Общие методические рекомендации по выполнению домашней контрольной работы

Для выполнения домашней контрольной работы необходимо изучить материал курса. Изучение ведется по программе и при использовании источников литературы, приведенных в данных методических рекомендациях.

Основным методом изучения учебного предмета является самостоятельная работа, которая должна проводиться в последовательности, предусмотренной программой учебного предмета, и быть обязательно систематической.

Задания на домашнюю контрольную работу №1 разработаны в количестве 100 вариантов в соответствии с программой курса. Номера заданий выбираются в соответствии с двумя последними цифрами шифра учащегося, на пересечении соответствующей строки с соответствующим столбцом из таблицы 1.

Каждый вариант содержит 4 задания: два теоретических вопроса и два практических задания.

Для выполнения домашней контрольной работы вначале изучается теория и приведенные примеры решения задач.

Первых два задания относятся к теоретическому содержанию учебного предмета и требуют ознакомления с соответствующими литературными источниками. Требуется раскрыть содержание теоретических вопросов. Привести примеры. Объем – около трех страниц.

Следующие два задания – задачи, в которых нужно составить программу на предложенном языке.

При оформлении домашней контрольной работы следует придерживаться следующих требований:

- работа выполняется в отдельной тетради или на листах А4 (шрифт 12-14, межстрочный интервал - одинарный). Следует пронумеровать страницы и оставить на них поля: справа – не менее 3 см для замечаний преподавателя, остальные поля – 2,5 см;
- на титульном листе указывается учебный предмет и номер работы, номер учебной группы, фамилия, имя, отчество учащегося, шифр;
- ответ следует начинать с номера и полного названия вопроса, он должен содержать схему алгоритма решения задачи, текст программы с краткими, но достаточно обоснованными, пояснениями;

- в конце работы следует указать список используемых источников, которым вы пользовались, проставить дату выполнения работы и подпись;

- учащиеся должны соблюдать абзац, всякую новую мысль учащийся должен начинать с новой строки;

- при вставке графических объектов или таблиц они должны быть пронумерованы в каждом вопросе отдельно или допускается сквозная нумерация на протяжении всей домашней контрольной работы. На рисунок или таблицу обязательно должна быть ссылка по тексту ответа на вопрос;

- не допускается использовать в работе авторские обороты;

- к домашней контрольной работе прикладывается диск с файлами, содержащими программы решения задачи;

- если в работе допущены недочеты или ошибки, то учащийся должен выполнить все указания преподавателя, сделанные в рецензии;

- домашняя контрольная работа должна быть выполнена в срок (в соответствии с учебным графиком);

- учащиеся, не имеющие зачета по домашней контрольной работе, к экзамену не допускаются;

- незачтенные работы исправляются в соответствии с замечаниями преподавателя и повторно сдаются для проверки на заочное отделение.

Требования к оформлению задач:

- для каждого файла задачи привести программный код;

- первой строкой программного кода должен быть комментарий, содержащий имя файла;

- чертежи и схемы следует выполнять аккуратно, соблюдая масштаб и ГОСТ19.701-90 (ИСО 5807-85);

- под программным кодом описать все теги, инструкции и функции, используемые в программе: их назначение, результат действия.

К домашней контрольной работе прикладывается диск с выполненными файлами. Каждая задача оформляется в отдельной папке. Имена папкам даются по номеру задачи.

Критерии оценки домашней контрольной работы

Домашняя контрольная работа, признанная преподавателем удовлетворительной и содержащая 75% положенного объема, оценивается отметкой «зачтено».

Домашняя контрольная работа оценивается отметкой «не зачтено», если:

- выполнена не в соответствии с вариантом;
- не раскрыто основное содержание теоретического вопроса (15%) и есть незначительные недочеты в практических заданиях (в сумме более 10%);
- не выполнено одно практическое задание (15%) и есть незначительные недочеты в остальных заданиях (в сумме более 10%);
- не выполнено два задания;
- нет диска с рабочими программами;
- есть существенные недочеты в нескольких практических заданиях (в сумме более 25%).

Учебная программа учебного предмета «Инструментальное программное обеспечение» и методические рекомендации по его изучению

Введение

Цели и задачи учебного предмета «Инструментальное программное обеспечение», связь с иными учебными предметами, значение в формировании профессиональных компетенций специалиста.

Классификационная характеристика средств разработки программного обеспечения.

Перспективы развития инструментальных средств разработки программного обеспечения.

Литература: [11] с. 3-7

Вопросы для самоконтроля

1 Перечислите разделы, на которые можно условно разделить изучаемый учебный предмет.

2 Назовите цели и задачи изучения учебного предмета «Инструментальное программное обеспечение».

3 Выскажите общее суждение о связи с иными учебными предметами, значении в формировании профессиональных компетенций специалиста.

4 Опишите классификационную характеристику средств разработки программного обеспечения.

5 Выскажите общее суждение о перспективах развития инструментальных средств разработки программного обеспечения.

Раздел 1 Синтаксис. Основные типы данных Python

Тема 1.1 Установка и знакомство с языком программирования Python

История создания языка программирования Python.

Возможности Python.

Текущее состояние языка. PEP8.

Среды разработки IDLE, PyCharm и иные.

Установка языка программирования Python.

Написание простейшей программы в среде разработки.
Литература: [11] с.7-14

Вопросы для самоконтроля

- 1 Изложите историю создания языка программирования Python, его возможности.
- 2 Опишите текущее состояние языка программирования Python.
- 3 Объясните PEP8, среды разработки IDLE, PyCharm и иные.
- 4 Опишите установку языка программирования Python.
- 5 Объясните написание простейшей программы в среде разработки.

Тема 1.2 Синтаксис Python. Типы данных в языке Python

Лексемы и литералы в Python. Отступы в Python. Комментарии и их типы. Идентификаторы Python.

Переменная. Объявление переменной и присвоение значений.

Объект. Ссылки на объект. Идентичность объекта. Получение помощи по объектам.

Имена переменных. Множественное присвоение. Типы переменных: локальная и глобальная. Удаление переменных. Максимально возможное значение переменной. Печать одиночных и множественных переменных.

Основные элементы и синтаксис языка программирования Python.

Форматирование python-кода.

Базовые типы данных Python.

Динамическая типизация в Python.

Литература: [11] с.15-27

Вопросы для самоконтроля

- 1 Опишите лексемы и литералы, отступы, комментарии и их типы, идентификаторы в языке программирования Python.
- 2 Дайте определение понятию «переменная» в языке программирования Python.
- 3 Объясните объявление переменной и присвоение значений в языке программирования Python.

4 Дайте определение понятию «объект» в языке программирования Python.

5 Опишите ссылки на объект, идентичность объекта, получение помощи по объектам в языке программирования Python.

6 Опишите имена переменных, множественное присвоение, типы переменных: локальная и глобальная, удаление переменных, максимально возможное значение переменной, печать одиночных и множественных переменных в языке программирования Python.

7 Опишите основные элементы и синтаксис языка программирования Python.

8 Объясните форматирование python-кода.

9 Назовите базовые типы данных в языке программирования Python.

10 Охарактеризуйте динамическую типизацию в языке программирования Python.

Тема 1.3 Числа в Python. Операторы и выражения

Целые числа. Вещественные числа. Комплексные числа.

Математические операции в Python. Приведение типов в Python. Операторы и выражения. Приоритет операторов.

Модули `math`, `decimal` и `random`. Тип `NoneType`. Тип данных «исключения (exceptions)».

Литература: [11], с. 27-33, 38-47

Вопросы для самоконтроля

1 Опишите целые числа, вещественные числа, комплексные числа в языке программирования Python.

2 Опишите математические операции, приведение типов, операторы и выражения, приоритет операторов в языке программирования Python.

3 Дайте определение понятиям модулям «`math`», «`decimal`» и «`random`» в языке программирования Python.

4 Объясните тип `NoneType`, тип данных «исключения (exceptions)» в языке программирования Python.

Тема 1.4 Логический (булевский) тип данных.

Операторы сравнения. Логические операторы

Характеристика логического типа данных.
Логические операторы. Операторы сравнения.
Литература: [11], с.33-38

Вопросы для самоконтроля

- 1 Опишите логический тип данных в языке программирования Python.
- 2 Объясните логические операторы и операторы сравнения в языке программирования Python.

Тема 1.5 Синтаксис инструкции if. Конструкция if...elif...else

Общий формат инструкции if. Простейшая форма инструкции if.
Управляющая конструкция if...else.
Ключевое слово elif. Конструкция (множественное ветвление) if...elif...else.
Литература: [11], с.48-53

Вопросы для самоконтроля

- 1 Опишите общий формат инструкции if, простейшую форму инструкции if в языке программирования Python.
- 2 Охарактеризуйте управляющую конструкцию if...else в языке программирования Python.
- 3 Опишите ключевое слово elif, конструкцию (множественное ветвление) if...elif...else в языке программирования Python.

Тема 1.6 Циклы while и for. Исключения

Цикл с условием (while).
Цикл обхода последовательности (for).
Алгоритмические приемы накопления суммы и произведения и их комбинация.
Счетчик событий. Досрочное завершение цикла (операторы break, continue и else). Поиск первого вхождения. Функция range().
Пользовательские исключения. Основы обработки и генерации исключений (унифицированный оператор try / except / finally, оператор

raise, оператор assert, оператор with / as). Встроенные классы исключений.

Литература: [11], с.54-66

Вопросы для самоконтроля

1 Объясните цикл с условием (while) в языке программирования Python.

2 Объясните цикл обхода последовательности (for) в языке программирования Python.

3 Охарактеризуйте алгоритмические приемы накопления суммы и произведения и их комбинацию в языке программирования Python.

4 Опишите счетчик событий, досрочное завершение цикла (операторы break, continue и else), поиск первого вхождения, функцию range() в языке программирования Python.

5 Назовите пользовательские исключения, основы обработки и генерации исключений (унифицированный оператор try / except / finally, оператор raise, оператор assert, оператор with / as), встроенные классы исключений в языке программирования Python.

Раздел 2 Структурированные типы данных

Тема 2.1 Строки

Понятие о строке. Создание строк. Литералы строк.

Экранированные последовательности. Операторы строк. Доступ по индексу и срезы строк.

Функции и методы строк. Форматирование строк. Кодировка для символов.

Литература: [11], с.67-86

Вопросы для самоконтроля

1 Опишите строку, создание строк, литералы строк в языке программирования Python.

2 Охарактеризуйте экранированные последовательности, операторы строк, доступ по индексу и срезы строк в языке программирования Python.

3 Объясните функции и методы строк, форматирование строк, кодировку для символов в языке программирования Python.

Тема 2.2 Списки

Понятие о списке. Создание списков.

Генератор списков. Доступ по индексу. Срезы списков. Операторы списков.

Функции и методы списков, и их отличие от строковых методов.

Многомерные списки.

Литература: [11], с.87-92

Вопросы для самоконтроля

- 1 Опишите список в языке программирования Python.
- 2 Объясните создание списков в языке программирования Python.
- 3 Опишите генератор списков, доступ по индексу, срезы списков, операторы списков в языке программирования Python.
- 4 Назовите функции и методы списков, и их отличие от строковых методов в языке программирования Python.
- 5 Дайте определение понятию «многомерные списки» в языке программирования Python.

Тема 2.3 Последовательность. Кортежи

Понятие о последовательности. Операции для последовательностей. Понятие кортежа. Создание кортежей. Доступ по индексу. Срезы кортежей. Операторы кортежей. Функции и методы кортежей. Отличие кортежа от списка.

Литература: [11], с.92-94

Вопросы для самоконтроля

- 1 Опишите последовательность, операции для последовательностей в языке программирования Python.
- 2 Опишите кортеж, создание кортежей, доступ по индексу, срезы кортежей, операторы кортежей в языке программирования Python.
- 3 Объясните функции и методы кортежей, отличие кортежей от списков в языке программирования Python.

Тема 2.4 Словари. Множества

Понятие о словаре. Создание словаря. Ключи словаря. Доступ к элементам словаря. Операторы словаря. Функции и методы словарей. Представление словарей.

Понятие множества. Создание множества. Операторы множества. Функции и методы множества. Фиксированные множества.

Литература: [11], с.94-102

Вопросы для самоконтроля

1 Опишите словарь, создание словаря, ключи словаря, доступ к элементам словаря, операторы словаря, функции и методы словарей в языке программирования Python.

2 Объясните представление словарей в языке программирования Python.

3 Опишите множество, создание множества, операторы множества, функции и методы множеств в языке программирования Python.

4 Объясните фиксированные множества в языке программирования Python.

Тема 2.5 Функции

Понятие функции. Объявление функций (def). Собственные (пользовательские) функции. Параметры и аргументы функции (обязательные и необязательные аргументы, именованные аргументы, произвольное количество аргументов). Области видимости (локальные и глобальные переменные).

Рекурсивный вызов функции. Лямбда-функции. Побочный эффект функции. Программирование сверху вниз.

Литература: [11], с.102-124

Вопросы для самоконтроля

1 Опишите функцию, объявление функций (def), собственные (пользовательские) функции, параметры и аргументы функции (обязательные и необязательные аргументы, именованные аргументы, произвольное количество аргументов), области видимости (локальные и глобальные переменные) в языке программирования Python.

2 Объясните рекурсивный вызов функции, лямбда-функции, побочный эффект функции, программирование сверху вниз в языке программирования Python.

Тема 2.6 Модули и пакеты

Модульное программирование.

Модули. Классификация модулей.

Пакеты. Подключение модулей и пакетов к программе с помощью: `import`, `from...import`. Относительный и абсолютный импорт.

Литература: [11], с.125-140

Вопросы для самоконтроля

1 Объясните модульное программирование в языке программирования Python.

2 Опишите модули в языке программирования Python.

3 Назовите классификацию модулей в языке программирования Python.

4 Опишите пакеты, подключение модулей и пакетов к программе с помощью: `import`, `from...import`, относительный и абсолютный импорт в языке программирования Python.

Тема 2.7 Файлы. Форматы

Понятие о файле. Открытие файла. Запись и чтение из текстового файла. Чтение из файла в двоичном режиме.

Функции для работы с файлами.

Связка операторов `with as`. Модули `struct` и `pickle`. Сериализация / десериализация. Форматы: CVS, JSON, XML, YAMI и INI. Извлечение информации.

Форматирование строк и запись в файл. Работа с операционной системой (модули `os`, `os.path` и `sys`).

Литература: [11], с. 140-171

Вопросы для самоконтроля

1 Опишите файл, открытие файла, запись и чтение из текстового файла, чтение из файла в двоичном режиме в языке программирования

Python.

2 Назовите функции для работы с файлами в языке программирования Python.

3 Опишите связку операторов with as, модули struct и pickle, сериализацию / десериализацию, форматы CVS, JSON, XML, YAMI и INI в языке программирования Python.

4 Объясните извлечение информации, форматирование строк и запись в файл, работу с операционной системой (модули os, os.path и sys) в языке программирования Python.

Раздел 3 Объектно-ориентированное программирование в Python

Тема 3.1 Основы объектно-ориентированного программирования. Объекты. Классы

Технология объектно-ориентированного программирования (ООП). Объекты и классы.

Объявление класса. Атрибуты класса. Поля класса.

Методы класса. Аргумент self. Уровни доступа атрибутов. Создание объекта. Конструкторы. Деструкторы. Удаление атрибутов и объектов.

Литература: [11], с.172-190

Вопросы для самоконтроля

1 Опишите технологию объектно-ориентированного программирования (ООП), объекты и классы в языке программирования Python.

2 Охарактеризуйте объявление класса, атрибуты класса, поля класса в языке программирования Python.

3 Опишите методы класса, аргумент self, уровни доступа атрибутов, создание объекта, конструкторы, деструкторы, удаление атрибутов и объектов в языке программирования Python.

Тема 3.2 Инкапсуляция. Наследование. Полиморфизм. Абстракция

Абстракция и ее назначение. Абстрактный метод.

Инкапсуляция и ее назначение.

Наследование. Синтаксис. Переопределение методов. Проверка наследования. Множественное и многоуровневое наследование. Особенности и назначение наследования.

Полиморфизм.

Литература: [11], с.201-208

Вопросы для самоконтроля

1 Опишите абстракцию и ее назначение, абстрактный метод в языке программирования Python.

2 Опишите инкапсуляцию и ее назначение в языке программирования Python.

3 Опишите наследование, синтаксис, переопределение методов, проверку наследования, множественное и многоуровневое наследование, особенности и назначение наследования в языке программирования Python.

4 Опишите полиморфизм в языке программирования Python.

Тема 3.4 Перегрузка операторов. Свойства и декораторы классов

Имитация типов (перегрузка операторов).

Перегрузка операторов как один из способов реализации полиморфизма.

Специальные функции.

Перегрузка арифметических операторов. Замыкания в Python.

Декораторы. Символ @. Документация декорируемых функций. Декораторы с аргументами. Классы-декораторы. Свойства классов.

Литература: [11], с. 190-201, 208-226

Вопросы для самоконтроля

1 Опишите имитацию типов (перегрузка операторов) в языке программирования Python.

2 Опишите перегрузку операторов как один из способов реализации полиморфизма в языке программирования Python.

3 Охарактеризуйте специальные функции в языке программирования Python.

4 Объясните перегрузку арифметических операторов,

замыкания в языке программирования Python.

5 Опишите декораторы, символ @, документацию декорируемых функций, декораторы с аргументами, классы-декораторы, свойства классов в языке программирования Python.

Раздел 4 Взаимодействие с сетью интернет.

Клиент- серверные приложения в Python.

Сервер-сокетные соединения

Тема 4.1 Разбор URL – адреса

Понятие о URL. Разбор URL-адреса.

Литература: [11], с.277-282

Вопросы для самоконтроля

1 Дайте определение понятию «URL-адрес» в языке программирования Python.

2 Объясните параметры и обработку URL-адреса в языке программирования Python.

3 Опишите извлечение параметров из URL-адреса с использованием возможностей модуля urlparse в языке программирования Python.

4 Опишите инструменты обработки url-адреса в языке программирования Python.

5 Объясните разбор URL-адреса в языке программирования Python.

Тема 4.2 Обмен данными по протоколу HTTP.

TCP/IP-протокол

Протокол HTTP. Обмен данными по протоколу HTTP.

Протокол TCP/IP. Клиент PuTTY. веб-браузер. веб-сервис reqbin.com.

Классы HTTPConnection, HTTPResponse.

Литература: [11], с.282-300

Вопросы для самоконтроля

- 1 Охарактеризуйте протокол HTTP, обмен данными по протоколу HTTP в языке программирования Python.
- 2 Опишите протокол TCP/IP, клиент PuTTY, веб-браузер, веб-сервис reqbin.com в языке программирования Python.
- 3 Опишите методы и атрибуты класса HTTPConnection для обмена данными в языке программирования Python.
- 4 Опишите методы и атрибуты класса HTTPResponse для обмена данными в языке программирования Python.

Тема 4.3 Сокеты. Клиентские сокеты. Серверные сокеты

Понятие о сокете (TCP-сокеты и UDP-сокеты).

Клиентские сокеты. Отправка информации на сервер. Серверные сокеты.

Получение и обработка информации от клиента. Отправка информации клиенту и закрытие сокета.

Литература: [11], с.300-308

Вопросы для самоконтроля

- 1 Опишите сокеты (TCP-сокеты и UDP-сокеты) в языке программирования Python.
- 2 Опишите клиентские сокеты, отправку информации на сервер, серверные сокеты в языке программирования Python.
- 3 Опишите получение и обработку информации от клиента, отправку информации клиенту и закрытие сокета в языке программирования Python.

Тема 4.4 Обмен данными с помощью модулей

Библиотека Requests. Установка библиотеки Requests. Импортирование модуля requests. Метод GET. Метод POST. Модуль urllib.

Открытие и чтение URL-адресов с помощью модуля urllib.request. Разбор URL-адресов с помощью модуля urllib.parse. Чтение заголовка

ответа HTML с помощью модуля `urllib.request`.

Обработка исключений с помощью модуля `urllib.request`.
Обработка исключений с помощью модуля `urllib.error.HTTPError`.

Литература: [11], с.308-329

Вопросы для самоконтроля

1 Опишите библиотеку `Requests`, методы GET и POST, модуль `urllib` в языке программирования Python.

2 Объясните установку библиотеки `Requests`, импортирование модуля `requests` в языке программирования Python.

3 Опишите открытие и чтение URL-адресов с помощью модуля `urllib.request`, разбор URL-адресов с помощью модуля `urllib.parse`, чтение заголовка ответа HTML с помощью модуля `urllib.request` в языке программирования Python.

4 Объясните обработку исключений с помощью модуля `urllib.request`, обработку исключений с помощью модуля `urllib.error.HTTPError` в языке программирования Python.

Раздел 5 Фреймворки и библиотеки языка Python

Тема 5.1 Развертывание фреймворка Django на персональном компьютере и начало работы

Фреймворк Django.

Выбор версии фреймворка Django. Установка и настройка фреймворка Django на персональном компьютере (далее – ПК). Структура приложений на Django. Создание базового проекта. Запуск приложения.

Модели данных Django. Административная панель Django Admin.

Литература: [8], с.14-105

Вопросы для самоконтроля

1 Дайте определение понятию «фреймворк Django» в языке программирования Python.

2 Объясните выбор версии фреймворка Django, установку и настройку фреймворка Django на персональном компьютере (далее – ПК), структуру приложений на Django, создание базового проекта и запуск приложения в языке программирования Python.

3 Опишите модель данных Django в языке программирования Python.

4 Объясните административную панель Django Admin в языке программирования Python.

Тема 5.2 Основы фреймворка Django

Понятие об основах фреймворка Django.

Маршрутизация. Представления и функции. Использование шаблонов. Структура и компоненты проекта. Правка созданного проекта.

Литература: [8], с.105-232

Вопросы для самоконтроля

1 Опишите основы фреймворка Django в языке программирования Python.

2 Объясните маршрутизацию, представление и функции, использование шаблонов, структуру и компоненты проекта, правку созданного проекта в языке программирования Python.

Тема 5.3 Работа с данными

Понятие о работе с данными. Обработка входящих запросов. Работа с формами. Отправка данных. Интеграция с базами данных. Публикация сайта в сети Интернет.

Литература: [8], с.232-450

Вопросы для самоконтроля

1 Опишите работу с данными в языке программирования Python.

2 Объясните обработку входящих запросов, работу с формами, отправку данных, интеграцию с базами данных в языке программирования Python.

3 Опишите публикацию сайта в сети Интернет в языке программирования Python.

Список используемых источников

Основная литература

- 1 Бэрри, Пол. Изучаем программирование на Python / Пол Бэрри. – М.: Эксмо, 2017. – 624 с.: ил.
- 2 Васильев, А. Н. Python на примерах. Практический курс по программированию / А.Н. Васильев. – Москва: Наука и техника, 2016. – 432 с.: ил.
- 3 Гуриков, С. Р. Основы алгоритмизации и программирования на Python: учебное пособие / С. Р. Гуриков. – Москва: ИНФРА-М, 2026. – 343 с.
- 4 Дронов, В. А. Django 4. Практика создания веб-сайтов на Python / В. А. Дронов. – СПб. : БХВ-Петербург, 2023. – 800 с.: ил.
- 5 Меле А. Django 4 в примерах / А. Меле; пер. с англ. А. В. Логунова. – М.: ДМК Пресс, 2023. – 800 с.: ил.
- 6 Нисчал Н. Python — это просто. Пошаговое руководство по программированию и анализу данных / Н. Нисчал. – СПб.: БХВ-Петербург, 2023. – 416 с.: ил.
- 7 Плас, Дж. В. Python для сложных задач: наука о данных и машинное обучение / Дж. В. Плас. – СПб.: Питер, 2018. – 576 с.: ил.
- 8 Постолиит, А. В. Python, Django и PyCharm для начинающих / А. В. Постолиит. – СПб.: БХВ-Петербург, 2021. – 464 с.: ил.
- 9 Постолиит, Анатолий. Разработка кроссплатформенных мобильных и настольных приложений на Python. Практическое пособие / Анатолий Постолиит. – Интеллектуальная издательская система Ridero, 2022. – 771 с.: ил.
- 10 Прохоренок, Н. А. Python 3. Самое необходимое / Н. А. Прохоренок, В. А. Дронов. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2019. – 608 с.: ил.
- 11 Фоминых, Е. И. Инструментальное программное обеспечение: учебное пособие / Е. И. Фоминых, Т. Е. Фоминых. – Минск : РИПО, 2022. – 410 с.: ил.
- 12 Фримен, Эрик. Учимся программировать с примерами на Python / Эрик Фримен. – СПб.: ООО «Диалектика», 2020. – 640 с.: ил.

Дополнительная литература

- 1 Бриггс, Джейсон. Python для детей. Самоучитель по программированию / Джейсон Бриггс; пер. с англ. Станислава

Ломакина ; [науч. ред. Д. Абрамова]. – Москва: Манн, Иванов и Фербер, 2017. – 320 с.

2 Вордерман, К. Программирование на Python: иллюстрированное руководство для детей / К. Вордерман, К. Стили, К. Квигли. – М.: Манн, Иванов и Фербер, 2018. – 224 с.

3 Златопольский, Д. М. Основы программирования на языке Python / Д. М. Златопольский. – М.: ДМК Пресс, 2017. – 284 с.

4 Лутц, М. Программирование на Python, том I, 4-е издание / М. Лутц. – СПб.: Символ-плюс, 2011. – 992 с.

5 Лутц, М. Программирование на Python, том II, 4-е издание / М. Лутц. – СПб.: Символ-плюс, 2011. – 992 с.

6 МакГрат, М. Программирование на Python для начинающих / М. МакГрат. – М.: Эксмо, 2015. – 192 с.

7 Медведев, М. А. Разработка веб-приложений на Django Framework: учебно-методическое пособие / М. А. Медведев, М. А. Медведева; Министерство науки и высшего образования Российской Федерации, Уральский федеральный университет. – Екатеринбург: Урал. ун-та, 2024. – 148 с.

8 Мэттиз, Э. Изучаем Python: программирование игр, визуализация данных, веб-приложения. 3-е изд. / Э. Мэттиз. – СПб.: Питер, 2020. – 512 с.

9 Саммерфилд, М. Программирование на Python 3. Подробное руководство / М. Саммерфилд. – СПб.: Символ-Плюс, 2009. – 608 с.

Задания на домашнюю контрольную работу №1 по учебному предмету «Инструментальное программное обеспечение»

1-60 Теоретические вопросы

- 1 Перечислите типы данных в языке программирования Python.
- 2 Приведите синтаксис и опишите алгоритм работы оператора ветвления `if...elif...else` в языке программирования Python.
- 3 Опишите лексемы и литералы, отступы, комментарии и их типы, идентификаторы в языке программирования Python.
- 4 Приведите синтаксис и опишите алгоритм работы оператора цикла `while` в языке программирования Python.
- 5 Опишите целые числа, вещественные числа, комплексные числа, математические операции, приведение типов, операторы и выражения, приоритет операторов в языке программирования Python.
- 6 Опишите счетчик событий, досрочное завершение цикла (операторы `break`, `continue` и `else`), поиск первого вхождения, функцию `range()` в языке программирования Python.
- 7 Объясните пользовательские исключения, основы обработки и генерации исключений (унифицированный оператор `try / except / finally`, оператор `raise`, оператор `assert`, оператор `with / as`), встроенные классы исключений в языке программирования Python.
- 8 Опишите строку, создание строк, литералы строк, экранированные последовательности, операторы строк, доступ по индексу и срезы строк.
- 9 Перечислите логические операторы и операторы сравнения в языке программирования Python.
- 10 Приведите синтаксис и опишите способы создания словаря и множества, ключи словаря, доступ к элементам словаря, операторы словаря и множества, функции и методы словарей и множеств в языке программирования Python.
- 11 Объясните функции и методы строк, форматирование строк, кодировку для символов в языке программирования Python.
- 12 Объясните тип `NoneType`, тип данных «исключения (exceptions)» в языке программирования Python.
- 13 Опишите файл, открытие файла, запись и чтение из текстового файла, чтение из файла в двоичном режиме в языке программирования Python.
- 14 Приведите синтаксис и опишите способы создания списков,

генератор списков, доступ по индексу, срезы списков, операторы списков в языке программирования Python.

15 Перечислите основные функции и константы модуля `math` в языке программирования Python.

16 Приведите синтаксис и опишите последовательность, операции для последовательностей, кортеж, создание кортежей, доступ по индексу, срезы кортежей, операторы кортежей в языке программирования Python.

17 Опишите ссылки на объект, идентичность объекта, получение помощи по объектам в языке программирования Python.

18 Опишите имена переменных, множественное присвоение, типы переменных: локальная и глобальная, удаление переменных, максимально возможное значение переменной, печать одиночных и множественных переменных в языке программирования Python.

19 Приведите синтаксис и опишите алгоритм работы цикл обхода последовательности (`for`) в языке программирования Python.

20 Опишите функции и методы списков, и их отличие от строковых методов в языке программирования Python.

21 Объясните функции и методы кортежей, отличие кортежей от списков в языке программирования Python.

22 Объясните представление словарей, фиксированные множества в языке программирования Python.

23 Опишите функцию, объявление функций (`def`), собственные (пользовательские) функции, параметры и аргументы функции (обязательные и необязательные аргументы, именованные аргументы, произвольное количество аргументов), области видимости (локальные и глобальные переменные) в языке программирования Python.

24 Объясните рекурсивный вызов функции, лямбда-функции, побочный эффект функции, программирование сверху вниз в языке программирования Python.

25 Опишите пакеты, подключение модулей и пакетов к программе с помощью: `import`, `from...import`, относительный и абсолютный импорт в языке программирования Python.

26 Объясните модульное программирование в языке программирования Python.

27 Опишите файл, открытие файла, запись и чтение из текстового файла, чтение из файла в двоичном режиме в языке программирования Python.

28 Охарактеризуйте функции для работы с файлами в языке

программирования Python.

29 Опишите связку операторов with as, модули struct и pickle, сериализацию / десериализацию, форматы CVS, JSON, XML, YAMI и INI в языке программирования Python.

30 Объясните извлечение информации, форматирование строк и запись в файл, работу с операционной системой (модули os, os.path и sys) в языке программирования Python.

31 Опишите технологию объектно-ориентированного программирования (ООП), объекты и классы в языке программирования Python.

32 Охарактеризуйте объявление класса, атрибуты класса, поля класса в языке программирования Python.

33 Опишите методы класса, аргумент self, уровни доступа атрибутов, создание объекта, конструкторы, деструкторы, удаление атрибутов и объектов в языке программирования Python.

34 Опишите абстракцию и ее назначение, абстрактный метод в языке программирования Python.

35 Опишите инкапсуляцию и ее назначение в языке программирования Python.

36 Опишите наследование, синтаксис, переопределение методов, проверку наследования, множественное и многоуровневое наследование, особенности и назначение наследования в языке программирования Python.

37 Опишите полиморфизм в языке программирования Python.

38 Опишите имитацию типов (перезагрузка операторов) в языке программирования Python.

39 Опишите перезагрузку операторов как один из способов реализации полиморфизма в языке программирования Python.

40 Охарактеризуйте специальные функции в языке программирования Python.

41 Объясните перезагрузку арифметических операторов, замыкания в языке программирования Python.

42 Опишите декораторы, символ @, документацию декорируемых функций, декораторы с аргументами, классы-декораторы, свойства классов в языке программирования Python.

43 Объясните разбор URL-адреса в языке программирования Python.

44 Охарактеризуйте протокол HTTP, обмен данными по протоколу HTTP в языке программирования Python.

45 Опишите протокол TCP/IP, клиент PuTTY, веб-браузер, веб-сервис reqbin.com в языке программирования Python.

46 Опишите методы и атрибуты класса HTTPConnection для обмена данными в языке программирования Python.

47 Опишите методы и атрибуты класса HTTPResponse для обмена данными в языке программирования Python.

48 Опишите сокет (TCP-сокеты и UDP-сокеты) в языке программирования Python.

49 Опишите клиентские сокеты, отправку информации на сервер, серверные сокеты в языке программирования Python.

50 Опишите получение и обработку информации от клиента, отправку информации клиенту и закрытие сокета в языке программирования Python.

51 Опишите библиотеку Requests, методы GET и POST, модуль urllib в языке программирования Python.

52 Объясните установку библиотеки Requests, импортирование модуля requests в языке программирования Python.

53 Опишите открытие и чтение URL-адресов с помощью модуля urllib.request, разбор URL-адресов с помощью модуля urllib.parse, чтение заголовка ответа HTML с помощью модуля urllib.request в языке программирования Python.

54 Объясните обработку исключений с помощью модуля urllib.request, обработку исключений с помощью модуля urllib.error.HTTPError в языке программирования Python.

55 Опишите модель данных Django в языке программирования Python.

56 Объясните административную панель Django Admin в языке программирования Python.

57 Опишите основы фреймворка Django в языке программирования Python.

58 Объясните маршрутизацию, представление и функции, использование шаблонов, структуру и компоненты проекта, правку созданного проекта в языке программирования Python.

59 Опишите работу с данными в языке программирования Python.

60 Объясните обработку входящих запросов, работу с формами, отправку данных, интеграцию с базами данных в языке программирования Python.

61-75 Практическое задание 1

Решение задач при помощи цикла While

В отчете необходимо представить текст программы с комментариями, блок-схему алгоритма решения, составленную в соответствии с ГОСТ 1.9.003-80 и 19.701-90 (ИСО 5807-85).

61 Напишите программу, вычисляющую сумму и среднее арифметическое последовательности положительных чисел, которые вводятся с клавиатуры.

62 Напишите программу, которая определяет максимальное число из введенной с клавиатуры последовательности положительных чисел (длина последовательности неограниченна).

63 Выведите на экран столько элементов ряда Фибоначчи, сколько указал пользователь. Например, если на ввод поступило число 6, то вывод должен содержать шесть первых чисел ряда Фибоначчи: 1 2 3 5 8 13.

64 Выведите на экран коды и символы таблицы ASCII, начиная с символа под номером 32 и заканчивая 127-м включительно. Вывод выполните в табличной форме: по десять пар «код-символ» в каждой строке.

65 Вычислите факториал введенного числа.

66 Выведите на экран ряд натуральных чисел от минимума до максимума с определенным шагом. Минимум, максимум и шаг указываются пользователем.

67 Найдите сумму n элементов следующего ряда чисел: 1 -0.5 0.25 -0.125 ... Количество элементов (n) вводится с клавиатуры.

68 Задано число a ($1 < a < 1.5$). Из чисел $1+1/2$, $1+1/3$, $1+1/4$..., напечатать те, которые не меньше a .

69 Даны два положительных числа K и N ($K < N$). Вывести все числа от K до N с помощью цикла while.

70 Напишите программу, которая выводит на экран таблицу значений функции $y = 2x^2 - 5x - 8$ в диапазоне от -3 до 3. Шаг изменения аргумента 0,5.

71 Напишите программу, которая выводит на экран таблицу значений функции $y = \sqrt{x^2 + 1}$ в диапазоне от -2 до 2. Шаг изменения аргумента 0,5.

72 Напишите программу, которая выводит на экран таблицу

значений функции $y = x^3 - 2x^2 + 8$ в диапазоне от -4 до 4. Шаг изменения аргумента 0,5.

73 Напишите программу, которая выводит на экран таблицу значений функции $y = \sqrt{|2 * x - 5|}$ в диапазоне от -2 до 2. Шаг изменения аргумента 0,1

74 Напишите программу, которая выводит на экран таблицу значений $y = \frac{1}{x^2 + 1}$ в диапазоне от -3 до 3. Шаг изменения аргумента 0,5.

75 Напишите программу, доказывающую или проверяющую, что для множества натуральных чисел выполняется равенство: $1+2+\dots+n = n(n+1)/2$, где n - любое натуральное число.

76- 100 Практическое задание 2

Решение задач обработки массивов

В отчете необходимо представить текст программы с комментариями, блок-схему алгоритма решения, составленную в соответствии с ГОСТ 29.003-80 и 19.701-90 (ИСО 5807-85).

76 Решите задачу. Дан двумерный массив. Заполните его по строкам с клавиатуры и определите:

- количество строк, не содержащих ни одного нулевого элемента;
- максимальное из чисел, в заданной строке массива.

77 Решите задачу. Дан двумерный массив. Заполните его по строкам с клавиатуры и определите:

- количество и сумму положительных элементов;
- минимальное из чисел, в заданной строке массива.

78 Решите задачу. Дан двумерный массив. Заполните его по строкам с клавиатуры и определите:

- количество столбцов, не содержащих ни одного нулевого элемента;
- минимальное из чисел, в заданном столбце массива.

79 Решите задачу. Дан двумерный массив. Заполните его по строкам с клавиатуры и определите:

- количество и сумму отрицательных элементов;

- максимальное из чисел, в заданном столбце массива

80 Решите задачу. Дан двумерный массив. Заполните его по строкам с клавиатуры и определите:

- количество строк, содержащих хотя бы один нулевой элемент;

- максимальное по модулю число, в заданной строке массива

81 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. Сформируйте список, в которой следует записать номера строк максимальных элементов каждого столбца.

82 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. В созданной последовательности найдите сумму элементов, сумма индексов которых равна 4.

83 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. В созданной последовательности найдите наибольший элемент побочной диагонали.

84 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 80. В созданной последовательности найдите четные элементы и выведите их как список.

85 Решите задачу. Дан двумерный массив $n \times n$. Заполните его по строкам с клавиатуры и определите:

- минимальный элемент в главной диагонали;

- произведение элементов в столбцах, не содержащих нулей.

86 Решите задачу. Дан двумерный массив $n \times n$. Заполните его по строкам с клавиатуры и определите:

- минимальный элемент в побочной диагонали;

- произведение элементов в строках, не содержащих нулей.

87 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от -15 до 40. В созданной последовательности вычислите среднее арифметическое положительных элементов и количество элементов равных нулю.

88 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. В созданной последовательности вычислите среднее арифметическое элементов, значение которых находится в заданном диапазоне. Диапазон задан значениями нижней и верхней границ, при этом

значения границ в диапазон не входят.

89 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 20. Сформируйте два списка, в один запишите элементы последовательности, расположенные на главной диагонали и выше, в другой – элементы последовательности, лежащие ниже главной диагонали, и выведите оба списка.

90 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 50. Выведите на экран только главную и побочную диагонали.

91 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. В результирующей последовательности выведите на экран половину последовательности относительно главной диагонали.

92 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. В созданной последовательности найдите сумму элементов на диагонали, параллельной побочной A_{14} , A_{23} , A_{32} , A_{41} , и сумму элементов на диагонали, параллельной главной A_{10} , A_{21} , A_{32} , A_{43} .

93 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. Преобразуйте вложенную последовательность таким образом, чтобы строки с нечетными индексами были упорядочены по убыванию, с четными – по возрастанию.

94 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. В созданной последовательности вычислите сумму наибольших значений в столбцах, выведите на экран список наибольших элементов.

95 Заполните вложенную последовательность $A[5,5]$ случайными целыми числами, находящимися в интервале от 1 до 40. Все элементы исходной последовательности замените нулями, а каждый элемент главной диагонали замените его номером.

96 Дана матрица $F(7,7)$. Найдите наименьший элемент в каждом столбце. Выведите матрицу и найденные элементы.

97 Найдите наибольший элемент главной диагонали матрицы $A(5,5)$ и выведите всю строку, в которой он находится.

98 Найдите наибольшие элементы каждой строки матрицы $Z(6,6)$ и поместить их на главную диагональ. Выведите полученную матрицу.

99 Вычислите суммы элементов матрицы $Y(3,3)$ по столбцам и выведите их.

100 Найдите наибольший элемент матрицы $A(10,10)$ и запишите нули в ту строку и столбец, где он находится. Выведите наибольший элемент, исходную и полученную матрицу.

Методические рекомендации по выполнению задач домашней контрольной работы № 1

Решение задач при помощи цикла While (задачи 61-75)

Условие задачи. Вводится последовательность вещественных чисел. Известно, что последний элемент последовательности равен пяти. Найдите количество положительных чисел и минимальное из них.

Решение:

Разработка алгоритма решения задачи представлена на рисунке 1.

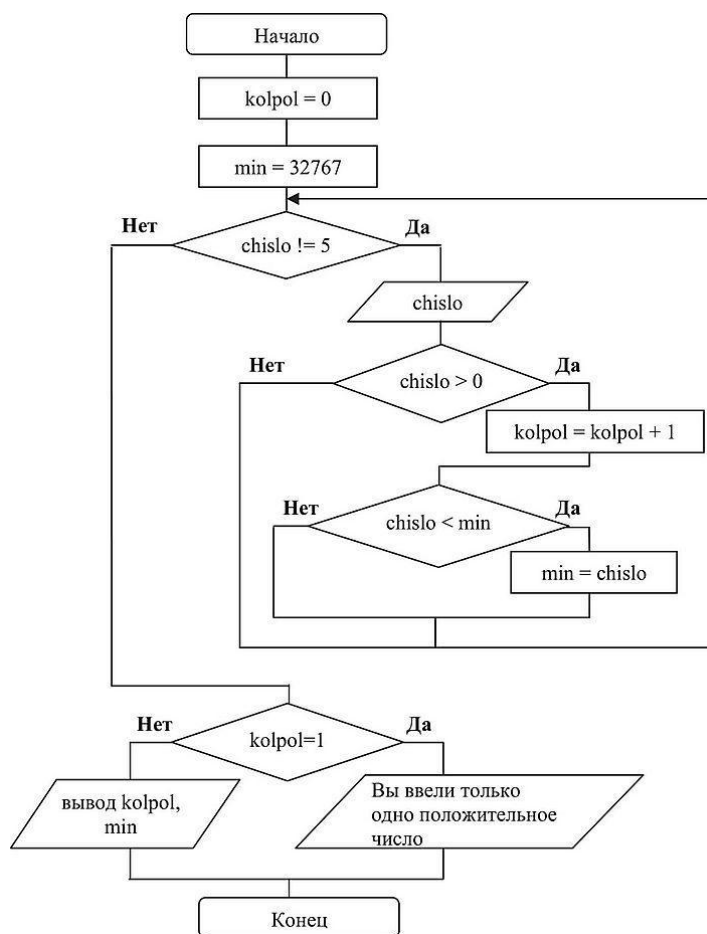


Рисунок 1 – Алгоритм решения задачи

Выход из цикла с оператором while осуществляется тогда, когда логическое выражение принимает значение False. Следовательно, до тех пор, пока пользователь вводит значения, не равные числу 5, в цикле будут выполняться соответствующие операторы, необходимые для поиска положительных чисел и минимального из них.

Текст программы приведен ниже.

```

kolpol=0
min=32767
chislo=0
while chislo!=5: #Пока число, введенное в числе, не равно числу 5,
выполнять тело цикла
    chislo=float(input("Введите число "))
    if chislo>0: #Проверка на положительность очередного
введенного числа
        kolpol= kolpol+1
        if chislo<min: #Реализация алгоритма поиска минимального
элемента
            min= chisl
    if kolpol==1:
        print("Вы ввели только одно положительное число: 5,
предназначенное для выхода из программы")
    else:
        print("Количество положительных чисел =", kolpol)
        print("Минимальное из них", min)

```

В результате выполнения выше приведенного кода будет получен результат:

```

Введите число  -5
Введите число  -3
Введите число   4
Введите число   5
Количество положительных чисел = 2
Минимальное из них 4.0

```

Решение задач обработки массивов (задачи 76 -100)

Условие задачи. Во вложенной последовательности произвольных чисел найдите максимальный элемент на главной диагонали и его номер (индекс).

Решение:

Разработка алгоритма решения задачи представлена на рисунке 2.

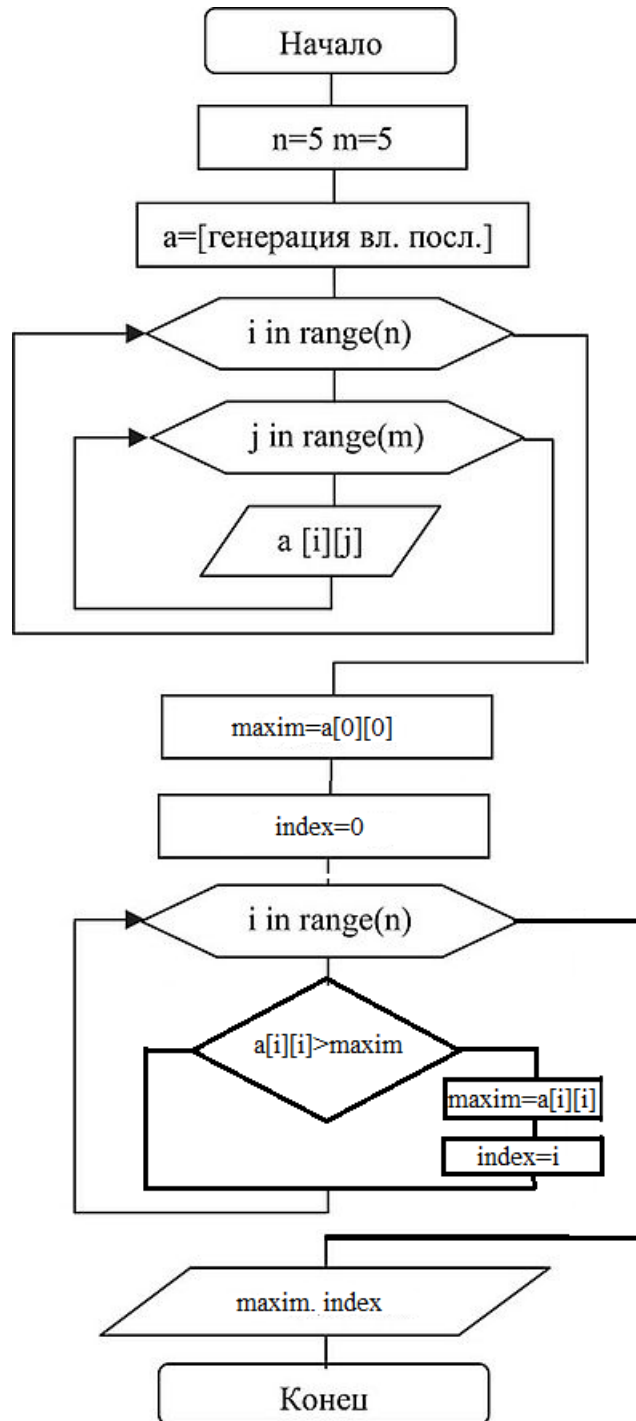


Рисунок 2 – Алгоритм решения задачи

Элементы главной диагонали рассматриваемой вложенной последовательности имеет следующие индексы: $a[0][0]$, $a[7][7]$, $a[5][5]$, $a[3][3]$, $a[4][4]$, поэтому для поиска максимального элемента применим следующий алгоритм решения задачи: максимальным считается нулевой элемент главной диагонали - $a[0][0]$. В программе за это действие отвечает оператор $\text{maxim} = a[0][0]$. Далее осуществляется сравнение очередного элемента главной диагонали с тем элементом,

который хранится в ячейке `maxim`, и, если он больше него, данный элемент заносится в ячейку `maxim` оператором `maxim=a[i][i]`.

Текст программы приведен ниже.

```
import random
n=5
m=5
a=[[i+j for j in random.sample(range(15),m)] for i in
random.sample(range(15),n)]
print("\n Исходная вложенная последовательность")
for i in range(n):
    for j in range(m):
        print (" %3d" %a[i][j], " ",end=" ")
    print()
maxim=a[0][0] #Максимальным считается нулевой элемент
главной диагонали
index=0
for i in range(n):
    if a[i][i]>maxim: #Поиск максимального элемента главной
диагонали
        maxim=a[i][i]
        index=i #Фиксация его номера
print("\n Максимальный элемент главной диагонали=", maxim)
print("\n Его индекс=", index)
```

В результате выполнения вышеприведенного кода будет получен результат:

15	12	11	14	20
13	11	16	21	25
13	7	16	6	9
17	14	15	26	25

Максимальный элемент главной диагонали= 25

Его индекс= 4

Таблица 1 – Варианты заданий на домашнюю контрольную работу № 1 по учебному предмету
«Инструментальное программное обеспечение»

Предпоследняя цифра шифра	Последняя цифра шифра									
	0	1	2	3	4	5	6	7	8	9
0	1, 60 67, 77	13, 57 74, 94	14, 38 61, 92	16, 58 63, 90	15, 60 70, 78	21, 55 71, 90	9, 54 74, 82	9, 43 64, 90	29, 32 68, 90	22, 56 70, 93
1	17, 52 61, 84	3, 42 73, 82	11, 35 75, 86	15, 39 67, 79	25, 51 67, 91	11, 33 61, 78	10, 44 62, 89	22, 35 69, 88	28, 48 64, 84	23, 57 73, 90
2	20, 48 73, 86	27, 49 62, 84	25, 54 65, 89	6, 50 74, 82	23, 38 66, 81	6, 58, 68, 88	18, 52 63, 87	16, 53 73, 76	12, 59 71, 82	12, 52 65, 76
3	16, 40 62, 77	19, 41 61, 85	30, 40 68, 79	20, 33 70, 85	2, 43 66, 76	3, 42 68, 78	25, 41 75, 81	13, 58 62, 88	29, 33 66, 80	13, 36 69, 83
4	21, 39 72, 82	5, 55 75, 87	22, 56 65, 87	8, 46 72, 77	5, 45 69, 76	16, 40 62, 88	14, 52 67, 78	14, 36 72, 82	4, 51 63, 83	4, 45 61, 85
5	30, 52 71, 83	24, 57 69, 86	28, 43 65, 83	28, 41 75, 80	3, 48 70, 84	20, 58 74, 77	6, 37 64, 77	26, 37 70, 81	21, 34 67, 83	30, 38 61, 84
6	23, 35 68, 87	5, 37 66, 86	18, 44 61, 96	21, 47 70, 87	7, 49 71, 79	8, 33 60, 78	24, 35 64, 84	21, 57 69, 97	4, 57 68, 87	24, 37 71, 89
7	15, 38 68, 90	7, 43 71, 79	17, 41 64, 88	27, 53 73, 85	10, 41 66, 77	16, 31 73, 76	7, 46 72, 88	19, 39 68, 77	9, 45 67, 98	27, 35 71, 80
8	26, 46 61, 80	8, 55 63, 79	8, 59 72, 80	1, 46 69, 88	1, 44, 75, 80	14, 57 75, 89	19, 39 66, 85	7, 42, 68, 83	22, 31 66, 81	26, 49 69, 86
9	17, 52 70, 81	29, 34 72, 87	14, 50 63, 99	25, 60 61, 83	18, 35 63, 79	7, 47 72, 95	2, 50 72, 81	2, 47 68, 78	3, 57 63, 85	8, 32 75, 82

