Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Уфимский государственный нефтяной технический университет»  
Кафедра «Вычислительная техника и инженерная кибернетика»  
  
  
  
  
  
Отчет  
по практике  
(тип: учебно-ознакомительная практика)

Студент гр. БПО09-24-02 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Хайруллин  
 подпись, дата  
  
Руководитель практики  
от кафедры ВТИК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.Г. Зайдуллина  
 подпись, дата, оценка

Уфа 2025

**1. Кодирование данных**

**1.Кодирование текстовых данных**

Выполнено кодирование своих ФИО на латинице и кириллице на Unicode (UTF-16 LE) и ASCII.

В ходе выполнения работы узнал, что Unicode - это универсальный стандарт для кодирования символов всех языков мира, а также множества специальных символов, поддерживает все языки, позволяя текстам на разных языках смешиваться без потери информативности. ASCII же представляет собой 7-битную кодировку, которая может представлять только базовые латинские буквы, цифры и управляющие символы, без поддержки кириллицы.

**2.Кодирование чисел**

Выполнено кодирование чисел в языке Python

В ходе выполнения работы узнал, что в Python целые числа представлены типом int с автоматическим расширением для чисел произвольной длины, вещественные - типом float, а также поддерживается тип complex для комплексных чисел, при этом в отличие от Java, C++ и PascalABC.NET Python использует динамическую типизацию и не требует явного указания типов переменных.

**2.Программирование**

**1.Стандартные алгоритмы**

Задание:

Задание III

1. Напишите программу перевода мегабайтов в биты, байты, килобайты, гигабайты. Результаты выводить на экран и в файл.

2. Модернизируйте алгоритм, чтобы он спрашивал переводимые единицы и осуществлял перевод. Диалог оформите следующим образом:

Единицы измерения:

1 – биты 2 – байты 3 – килобайты 4 – мегабайты 5 - гигабайты

Выберите единицы, **которые** переводить – *4*

Выберите единицы, **в которые** переводить – *2*

Введите переводимое значение – *2*

2 мегабайта = 2097152 байта

3. Доработайте предыдущий алгоритм так, чтобы система проверяла умение пользователя переводить единицы измерения. Предусмотрите многократную проверку перевода с подсчётом правильных ответов.

Единицы измерения:

1 – биты 2 – байты 3 – килобайты 4 – мегабайты 5 - гигабайты

Выберите единицы, **которые** переводить – *4*

Выберите единицы, **в которые** переводить – *2*

Введите переводимое значение – *2*

Ваш перевод - 2097152

Продолжить? (y/n) n

ВЕРНЫХ ОТВЕТОВ – 1

УДАЧИ!

Вывод:

Была разработана программа для конвертации единиц измерения информации (биты, байты, килобайты, мегабайты, гигабайты) с функцией проверки знаний пользователя. В ходе работы я закрепил навыки работы с файлами в Python, включая запись и чтение данных. Также закрепил умение создавать интерактивный интерфейс для взаимодействия с пользователем. Программа не только выполняет перевод между единицами измерения, но и проверяет правильность ответов пользователя, подсчитывая количество верных решений.

**2.Одномерные массивы данных**

Задание:

22 вариант.

1. 

2. Вычислить среднеарифметическое значение элементов последней трети массива

3. Определить наибольший по абсолютной величине среди каждых пятых

элементов второй половины массива

4. Упорядочить каждые третьи элементы последней трети массива по возрастанию обратных значений

Вывод:

Была написана программа решения четырех последовательных задач обработки одномерного массива данных.

В ходе выполнения работы я закрепил знания по поиску, чтению, записи и сортировки данных одномерного массива.

**3.Двумерные массивы данных. Подпрограммы**

Задание:

Задание. Напишите алгоритм формирования и вывода элементов двумерного массива данных (целочисленной матрицы Мn × m) генератором псевдослучайных чисел с последующим вызовом к нему **функций**

а) формирования на его основе одномерного массива (вектора V) по определённому правилу;

б) преобразования самой матрицы М\*;

в) матричной операции над вектором (V) и матрицей (M или М\*).

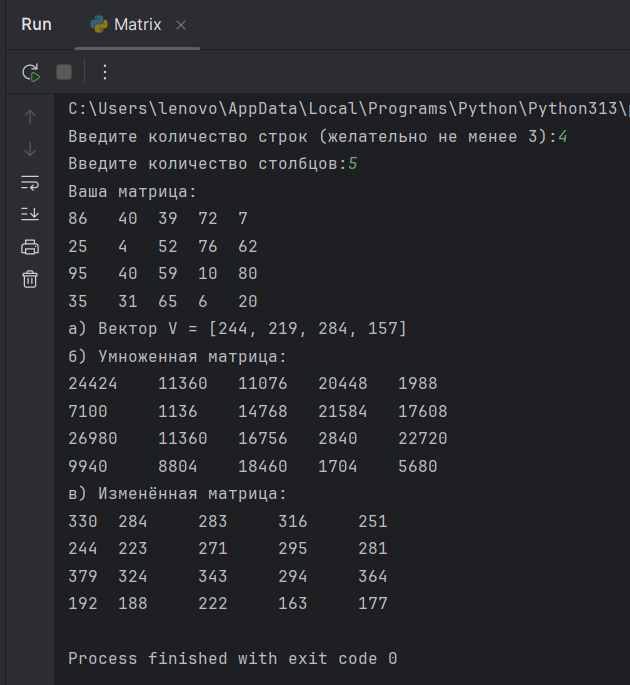
**22 вариант.**

а) Vk - сумма элементов k-строки матрицы;

б) умножить матрицу на третий элемент вектора;

в) каждый столбец матрицы сложить с вектором V.

Результат работы программы:

****

Вывод:

Был написан алгоритм формирования и вывода элементов двумерного массива данных генератором псевдослучайных чисел с последующим вызовом к нему функций.

В ходе выполнения работы я закрепил знания о двумерных массивах и векторах в рамках языка Python.

**4.Динамические массивы данных**

Было выполнено 5 последовательных заданий.

В ходе выполнения этих пяти я закрепил все ранее полученные знания из курса теории, а также о динамических массивах данных в рамках языка Python.

**5. Программы обработки строковых данных**

Задание:

Вариант 8. В массиве хранятся фамилии с инициалами учеников класса (Иванов П.П.). Требуется напечатать список класса с указанием для каждого ученика количества его однофамильцев.

Вывод:

Введённые инициалы учеников обработаны, подсчитано количество однофамильцев для каждого. Результаты выведены в формате: число однофамильцев — для каждого ученика с указанием его инициалов.

В ходе выполнения работы я закрепил навыки обработки строк и работы со списками в Python, а также применил методы подсчёта элементов и форматированного вывода результатов.

**6.Файлы и строки**

Задание:

Создать программу, позволяющую из одного текстового файла, содержащего несколько строк (тип String) текста на русском языке, построчно переписать в другой текстовый файл слова, отвечающие некоторому условию. Задания выполнять согласно вариантам.

Требования:

– в новом файле следует указать номер строки, в которой искомые слова находились в исходном файле;

– для каждой строки в конце указать количество выбранных слов.

Вариант 8. Переписать в результирующий файл слова с числом букв больше 5.

Вывод:

Создана программа, позволяющая из одного текстового файла, содержащего несколько строк (тип String) текста на русском языке, построчно переписать в другой текстовый файл слова с числом букв больше 5.

В ходе выполнения работы я закрепил знания о работе с файлами и строками в рамках языка Python, а также отработал навыки обработки текстовых данных и записи результатов в новый файл с форматированием.

**7. Сортировка массивов данных**

Задание:

В магазине для упаковки подарков есть N кубических коробок разной стоимости. Самой интересной считается упаковка подарка по принципу матрёшки: подарок упаковывается в одну из коробок, та в свою очередь в другую коробку и т. д. Одну коробку можно поместить в другую, если длина её стороны хотя бы на K единиц меньше длины стороны другой коробки. Определите наибольшее количество коробок, которое можно использовать для упаковки одного подарка, так чтобы стоимость упаковки не превысила M единиц, и минимальную итоговую стоимость этой упаковки Размер подарка позволяет поместить его в самую маленькую коробку. Если есть несколько вариантов упаковки с одинаковым наибольшим количеством коробок, выберите вариант с наименьшей стоимостью.  
Входные данные представлены в файле 8.txt следующим образом. В первой строке входного файла находится число N – количество коробок в магазине (натуральное число, не превышающее 10 000), число K – минимально допустимая разница длин сторон соседних коробок в матрёшке, и число M – максимально допустимая стоимость упаковки. В каждой из следующих N строк записаны длина стороны коробки (натуральное число, не превышающее 10 000) и стоимость коробки.  
Запишите в ответе два целых числа: наибольшее количество коробок, которое можно использовать для упаковки одного подарка, так чтобы стоимость упаковки не превысила M единиц, и минимальную итоговую стоимость упаковки.  
Пример входного файла:

5 3 10  
50 5  
40 6  
30 5  
20 3  
10 15

При таких исходных данных максимальное количество коробок (2) при минимальной стоимости (8) получается при использовании коробок со сторонами 50 и 20. Ответ: 2 8.

Вывод:

Создана программа, позволяющая определить максимальное количество коробок, которые можно выбрать при заданных ограничениях, и минимальную стоимость такого набора.

В ходе выполнения работы я закрепил знания об обработке и сортировке массивов данных, а также алгоритмах поиска оптимальных решений в рамках языка Python.

**3.Вывод по практике**

В ходе ознакомительной практики я закрепил и применил базовые концепции языка Python. На практике я освоил работу с переменными, используя динамическую типизацию — тип данных определяется автоматически при присваивании значения без явного указания. Я изучил управление ходом программы с помощью условных операторов (if/elif/else/match) и циклов (for/while), применяя их для решения математических задач и реализации полезных функций.

Особый акцент я сделал на изучении встроенных структур данных: списков, кортежей, словарей и множеств. Это позволило мне писать более сложные программы для обработки данных. Я также освоил специфические возможности Python, такие как списковые включения, срезы и динамическое изменение коллекций. Практика помогла мне лучше понять основы программирования на Python и показала его гибкость в решении различных задач.