**ЛАБОРАТОРНАЯ РАБОТА №5.**

**РАБОТА С ФАЙЛАМИ В PYTHON.**

**Цель работы.**

1. изучить особенности реализации чтения и записи файлов в Python;;
2. создать приложение, демонстрирующее различные способы работы с файлами в Python.

**Краткие теоретические сведения**

**Вывод данных в консоль**

Один из самых распространенных способов вывести данные в *Python* – это напечатать их в консоли. Если вы находитесь на этапе изучения языка, такой способ является основным для того, чтобы быстро просмотреть результат свой работы. Для вывода данных в консоль используется функция *print*.

Рассмотрим основные способы использования данной функции.

>>> print("Hello")

Hello

>>> print("Hello, " + "world!")

Hello, world!

>>> print("Age: " + str(**23**))

Age: 23

По умолчанию, для разделения элементов в функции *print* используется пробел.

>>> print("A", "B", "C")

A B C

Для замены разделителя необходимо использовать параметр *sep* функции *print*.

print("A", "B", "C", sep="#")

A#B#C

В качестве конечного элемента выводимой строки, используется символ перевода строки.

>>> for i in range(**3**):

print("i: " + str(i))

i: 0

i: 1

i: 2

Для его замены используется параметр *end*.

>>> for i in range(**3**):

print("[i: " + str(i) + "]", end=" -- ")

[i: 0] -- [i: 1] -- [i: 2] --

Ввод данных с клавиатуры

Для считывания вводимых с клавиатуры данных используется функция *input()*.

>>> input()

test

'test'

Для сохранения данных в переменной используется следующий синтаксис.

>>> a = input()

hello

>>> print(a)

hello

Если считывается с клавиатуры целое число, то строку, получаемую с помощью функции *input()*, можно передать сразу в функцию *int()*.

>>> val = int(input())

123

>>> print(val)

123

>>> type(val)

<*class* 'int'>

Для вывода строки-приглашения, используйте ее в качестве аргумента функции *input()*.

>>> tv = int(input("input number: "))

input number: 334

>>> print(tv)

334

Преобразование строки в список осуществляется с помощью метода *split()*, по умолчанию, в качестве разделителя, используется пробел.

>>> l = input().split()

1 2 3 4 5 6 7

>>> print(l)

['1', '2', '3', '4', '5', '6', '7']

Разделитель можно заменить, указав его в качестве аргумента метода*split()*.

>>> nl = input().split("-")

1-2-3-4-5-6-7

>>> print(nl)

['1', '2', '3', '4', '5', '6', '7']

Для считывания списка чисел с одновременным приведением их к типу *int* можно воспользоваться вот такой конструкцией.

>>> nums = map(int, input().split())

1 2 3 4 5 6 7

>>> print(list(nums))

[1, 2, 3, 4, 5, 6, 7]

**Работа с файлами**

**Открытие и закрытие файла**

Для открытия файла используется функция *open()*, которая возвращает файловый объект. Наиболее часто используемый вид данной функции выглядит так *open(имя\_файла, режим\_доступа)*.

Для указания режима доступа используется следующие символы:

‘*r*’ – открыть файл для чтения;

‘*w*’ – открыть файл для записи;

‘*x*’ – открыть файл с целью создания, если файл существует, то вызов функции *open* завершится с ошибкой;

‘*a*’ – открыть файл для записи, при этом новые данные будут добавлены в конец файла, без удаления существующих;

‘*b*’ – бинарный режим;

‘*t*’ – текстовый режим;

‘*+*’ – открывает файл для обновления.

По умолчанию файл открывается на чтение в текстовом режиме.

У файлового объекта есть следующие атрибуты.

*file.closed* – возвращает *true* если файл закрыт и *false* в противном случае;

*file.mode* – возвращает режим доступа к файлу, при этом файл должен быть открыт;

*file.name* – имя файла.

>>> f = open("test.txt", "r")

>>> print("file.closed: " + str(f.closed))

file.closed: False

>>> print("file.mode: " + f.mode)

file.mode: r

>>> print("file.name: " + f.name)

file.name: test.txt

Для закрытия файла используется метод *close()*.

**Чтение данных из файла**

Чтение данных из файла осуществляется с помощью методов *read(размер)* и *readline()*.

Метод *read(размер)* считывает из файла определенное количество символов, переданное в качестве аргумента. Если использовать этот метод без аргументов, то будет считан весь файл.

>>> f = open("test.txt", "r")

>>> f.read()

'1 2 3 4 5**\n**Work with file**\n**'

>>> f.close()

В качестве аргумента метода можно передать количество символом, которое нужно считать.

>>> f = open("test.txt", "r")

>>> f.read(**5**)

'1 2 3'

>>> f.close()

Метод *readline()* позволяет считать строку из открытого файла.

>>> f = open("test.txt", "r")

>>> f.readline()

'1 2 3 4 5**\n**'

>>> f.close()

Построчное считывание можно организовать с  помощью оператора *for*.

>>> f = open("test.txt", "r")

>>> for line in f:

... print(line)

...

1 2 3 4 5

Work with file

>>> f.close()

**Запись данных в файл**

Для записи данных файл используется метод *write(строка)*, при успешной записи он вернет количество записанных символов.

>>> f = open("test.txt", "a")

>>> f.write("Test string")

11

>>> f.close()

**Дополнительные методы для работы с файлами**

Метод *tell()* возвращает текущую позицию “условного курсора” в файле. Например, если вы считали пять символов, то “курсор” будет установлен в позицию 5.

>>> f = open("test.txt", "r")

>>> f.read(**5**)

'1 2 3'

>>> f.tell()

5

>>> f.close()

Метод *seek(позиция)* выставляет позицию в файле.

>>> f = open("test.txt", "r")

>>> f.tell()

0

>>> f.seek(**8**)

8

>>> f.read(**1**)

'5'

>>> f.tell()

9

>>> f.close()

Хорошей практикой при работе с файлами является применение оператора *with*. При его использовании нет необходимости закрывать файл, при завершении работы с ним, эта операция будет выполнена автоматически.

>>> with open("test.txt", "r") as f:

... for line in f:

... print(line)

...

1 2 3 4 5

Work with file

Test string

>>> f.closed

True

**Задание 1: (работа с байтовыми потоками) (Номер задачи совпадает с номером варианта)**

1. Дан файл f, компоненты которого являются целыми числами. Записать в файл g все четные числа файла f, а в файл h-все нечетные. Порядок следования чисел сохранить
2. Дан файл, содержащий вещественные числа. Описать функцию, подсчитывающую сумму отрицательных элементов в этом файле.
3. Даны два файла, содержащих целые числа. Переписать содержимое первого файла во второй без отрицательных чисел.
4. Дан файл f, содержащий вещественные числа. Описать функцию less(f) от непустого файла f, меньших среднего арифметического всех элементов этого файла.
5. Дан файл вещественных чисел f. Определить количество элементов в наиболее длинной возрастающей последовательности файла f.
6. Дан файл f, компоненты которого являются целыми числами. Записать в файл g наибольшее значение первых пяти компонент файла f, затем следующих пяти компонент и т.д. Если в последней группе окажется менее пяти компонент, то последняя компонента файла g должна быть равна наибольшей из компонент файла f, образующих последнюю (неполную) группу.
7. Дан файл f, компоненты которого являются целыми числами. Получить файл g, образованный из файла f исключением повторных вхождений одного и того же числа.
8. Дан файл f, содержащий целые числа. Создать новый файл g и записать в него все простые числа файла f.
9. Дан файл целых чисел. Требуется переписать его содержимое в новый файл без автоморфных чисел
10. Дан файл целых чисел. Требуется определить есть ли в нем 3 подряд идущих простых числа, после которых следует совершенное число.
11. Дан файл целых чисел. Найти количество элементов файла, расположенных между максимальным и минимальным элементами файла12.Дан файл целых чисел. Переписать в новый файл четные элементы, расположенные между первым нечетным элементом и предпоследним четным элементов файл в.
12. Дан файл целых чисел. Переписать в новый файл все элементы, кратные минимальному нечетному элементу файла.
13. Дан файл целых чисел. Переписать в новый файл все элементы исходного файла, расположенные до первого простого элемента и после максимального элемента.
14. Даны 2 файла целых чисел. Требуется записать в третий файл попарные суммы элементов. Конечный файл должен содержать число элементов в коротком файле (если исходные файлы имеют различное число элементов)

**Задание 2 (Работа с символьными потоками) (Номер задачи совпадает с номером варианта)**

1. Дан текстовый файл f. В каждой строке найти позицию самого длинного слово среди слов, вторая буква которых «e»; если слов с наибольшей длиной несколько, то найти последнее. Результат сохранить в отдельный файл. Если таких слов нет вообще, то сообщить об этом.
2. Дан текстовый файл f. Записать "в перевернутом виде" строки файла f в файл g, оставляя только строки, в которых записано не менее двух предложений. Порядок строк в файле g должен совпадать с порядком исходных строк в файле f;
3. Дан текстовый файл f, содержащий сведения о сотрудниках учреждения, записанные по следующему образцу: фамилия имя отчество фамилия имя отчество... Записать эти сведения в файле g, используя образец:

фамилия и. о. фамилия и. о. ...;

1. Дан текстовый файл, в каждой строке котором записана следующая информация о студентах: имя студента, оценка по текущему контролю и оценка по рубежному контролю. Найти лучшего и худшего по успеваемости студентов группы. Определить средний балл по рейтингу в группе. Всех студентов, рейтинг которых выше среднего записать в новый файл.
2. Дан текстовый файл, в каждой строке котором записана следующая информация о студентах: имя студента, количество пропущенных часов, средний балл по итогам рейтингов. Получить новый файл, в котором записаны студенты, имеющие более 20 часов пропущенных занятий. Найти лучшего и худшего по успеваемости студентов группы.
3. Дан текстовый файл f, содержащий сведения о веществах: указывается название вещества, его удельный вес и проводимость (проводник, полупроводник, изолятор). Найти удельные веса и названия всех полупроводников. Переписать в отдельный файл вещества, имеющие максимальную проводимость.
4. Дан текстовый файл f, содержащий сведения о книгах. Сведения о каждой из книг - это фамилия автора, название и год издания. 1) Найти названия книг заданного автора, изданных с 1960 г. 2) Определить год последнего издания книги с названием "Информатика". Записать в отдельный файл список книг, изданных за последние 5 лет.
5. Дан текстовый файл f, содержащий сведения об экспортируемых товарах: указывается наименование товара, страна, импортирующая товар, и объем поставляемой партии в штуках. Найти страны, в которые экспортируется данный товар, и общий объем его экспорта. Найти самый востребованный товар (максимальный суммарный объем партии). 9. Дан текстовый файл. Описать процедуру, которая записывает в новый текстовый файл все цифры из исходного файла (разбиение на строки сохранить). В новом файле определить максимальное число, образованное выписанными цифрами.10.Пусть текстовый файл t разбит на непустые строки. Описать функции: 1) для подсчета числа строк, которые состоят из одинаковых литер; 2) для подсчета строк, в которых нет гласных; 3) для определения, какая из гласных букв встречается в нем чаще всего.

11.Пусть текстовый файл t разбит на непустые строки. Описать функции:1) для подсчета числа строк, являющимися предложениями (начинаются с заглавной буквы, и заканчиваются «.», «!» или «?»); 2) для определения строки, в которой наименьшее количество цифр и знаков пунктуации.

1. Описать функцию, которая находит максимальную длину строк текстового файла t. Переписать содержимое файла в другой файл без строк, длина которых совпадает с максимальной. Оставшиеся строки записать в третий файл без гласных букв и знаков препинания.
2. Дан текстовый файл f. Определить, есть ли в файле строки, начинающиеся и заканчивающиеся цифрами. Если да, то определить максимальное из чисел, которые можно составить из этих цифр (отдельно для каждой строки).
3. Дан текстовый файл g. Создать новый файл f и записать в него все слова файла g с количеством символов >4 и < 10, в которых нет повторяющихся символов. Если таких слов нет, то полностью переписать файл g в файл f.
4. Дан текстовый файл f. Создать новый файл g и записать в него построчно все слова файла f, предварительно перенеся последнюю букву слова в начало.

**Контрольные вопросы:**

* 1. Какие методы работы с файлами вы знаете?
  2. Какие параметры можно указать при открытии файла?
  3. Приведите пример построчного чтения текстового файла.
  4. Какая ошибка может возникнуть, если открываемый файл не найден?

Как ее обработать?