Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский  
политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчёт по лабораторной работе № 3  
тема «Строки, файлы»  
по дисциплине «Информатика»

Выполнил: студент группы ПМ-23-1б Кузнецов Д.В.

Проверил: ст. пр. каф. ВММБ Ильиных Г.В.

Пермь, 2023

# Задание 1

## Постановка задачи

Написать функцию *print\_char\_list(array)*, которая в цикле выводит в консоль элементы одномерного массива символов array.

Написать функцию task1(), в которой:

1. Задана функция, в которой создаётся одномерный массив символов из n элементов. Массив заполняется произвольными символами (буквами и цифрами).
2. Вызывается функция print\_char\_list для вывода полученного массива в консоль.
3. В цикле цифры заменяются символами «\*» и ещё раз вызывается функция print\_char\_list.

## Алгоритм решения

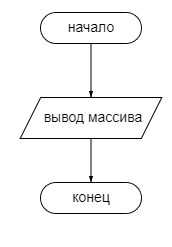


Рис.1 – Блок-схема функции *print\_char\_list(array)*

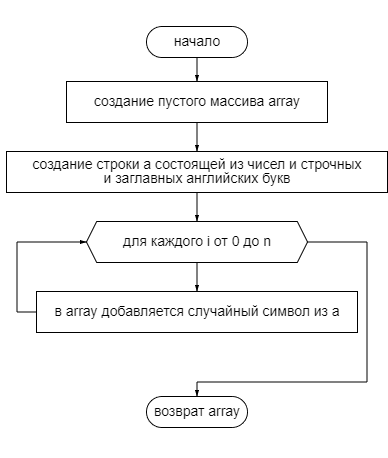


Рис.2 – Блок-схема *create\_random\_array(n)*

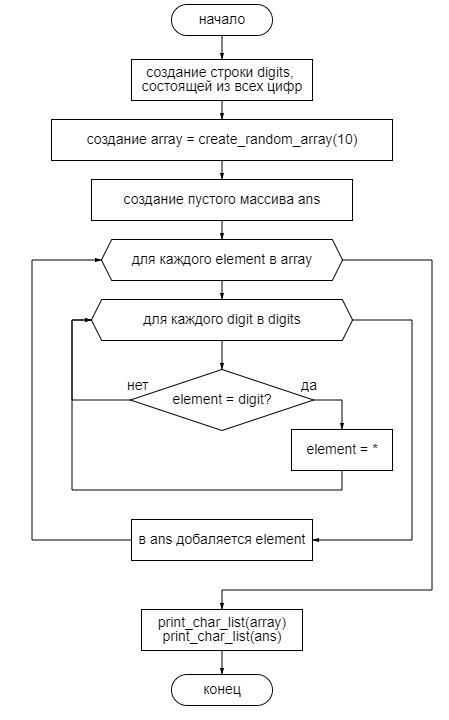


Рис.3 – Блок-схема программы №1

## Решение задачи, код программы

import random  
import string

def task1():  
 def print\_char\_list(array):  
 print(\*array)  
   
 def create\_random\_array(n):  
 array = []  
 a = string.ascii\_letters + string.digits  
 for i in range(n):  
 array.append(random.choice(a))  
 return array  
 digits = string.digits  
 array = create\_random\_array(10)  
 ans = []  
 for element in array:  
 for digit in digits:  
 if element == digit:  
 element = "\*"  
 ans.append(element)  
 print\_char\_list(array)  
 print\_char\_list(ans)

## Тестирование работы программы с проверкой

|  |  |
| --- | --- |
| №п.п | Решение Python |
| 1 |  |
| 2 |  |
| 3 |  |

# Задание 3

## Постановка задачи

## Написать функцию task3(), в которой:

## 1. Выводится сообщение: Введите текст:

## 2. Запрашивается ввод строковой переменной.

## 3. Переменная дозаписывается в файл, называющийся «a-words.dat», если она начинается с буквы «a», аналогично для других букв. Если слово начинается не с буквы, то запись не производится.

## 4.шаги 1 - 3 повторяются 5 раз.

## Алгоритм решения

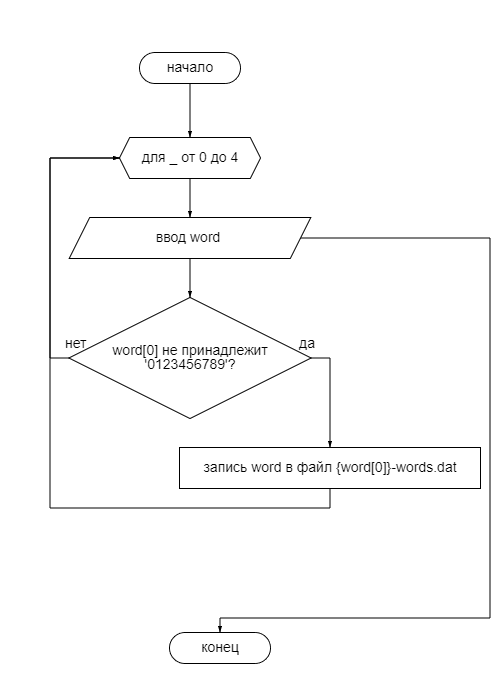


Рис.4 – Блок-схема программы №3

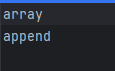
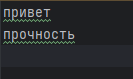
## Решение задачи, код программы

def task3():  
 for \_ in range(5):  
 word = str(input('Введите слово '))  
 if word[0] not in '0123456789':  
 with open(word[0] + "-words.dat", 'a') as file:  
 file.write(word + '\n')

## Тестирование работы программы с проверкой

Таблица 2 – Тестирование работы программы №3

|  |  |
| --- | --- |
| № п.п | Решение Python |
| 1 |  |
| 2 |  |
| 3 |  |

В результате работы программы были созданы файлы, в некоторые файлы были записаны несколько слов.  
  , , , 

# Задание 4

## Постановка задачи

Написать функцию *print\_string\_list(array)*, которая в цикле выводит в консоль элементы одномерного массива строк array. Написать функцию *task4()* , в которой:

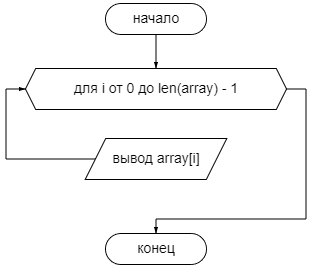
1. Задан массив строк, как минимум, из 10 элементов. Каждый элемент массива представляет собой одно слово или короткую фразу из 2 - 4 слов. Слова (фразы) должны представлять собой объекты какого-либо множества (адреса сайтов, названия приложений, названия социальных сетей, термины какой-либо учебной дисциплины и т.п.).

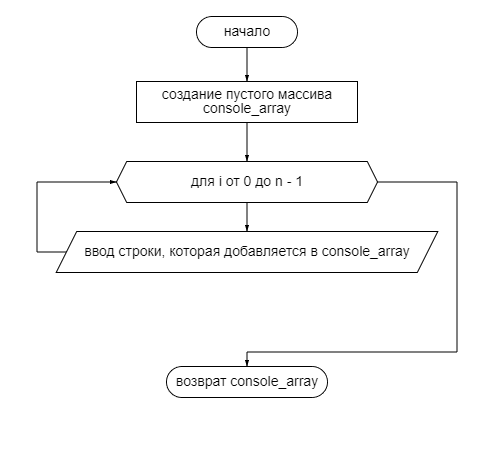
2. Задана функция *get\_string\_array\_from\_console()*, получающая в цикле с консоли элементы одномерного массива строк array и возвращающая этот массив.

3. Задана функция proverka(array1, array2), которая проверяет, совпадают ли элементы массива array1 с элементами массива array2. Если совпадения есть, то к соответствующим элементам массива array1 добавляются два символа: пробел и символ галочки.

4. С помощью *get\_string\_array\_from\_console()* у пользователя запрашивается одномерный массив строк array. 5. Для этого массива вызывается функция proverka. 6. С помощью *print\_string\_list* выводится модифицированный массив array.

## Алгоритм решения

  
Рис.5 – Код функции *print\_string\_list(array)*

Рис.6 – Блок-схема функции *get\_string\_array\_from\_console(n)*

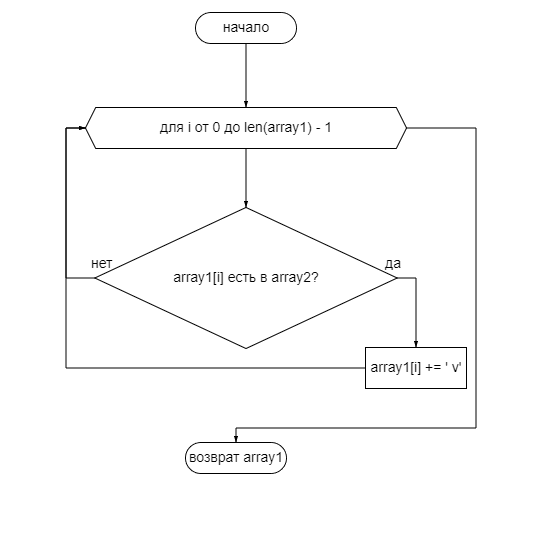


Рис.7 – Блок-схема функции *proverka(array1, array2)*

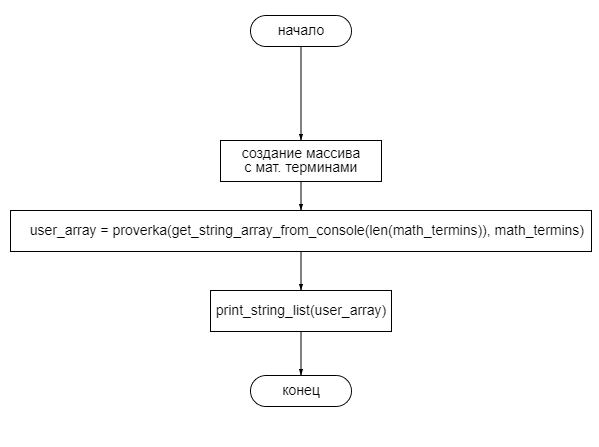


Рис.8 – Блок-схема программы №4

## Решение задачи, код программы

def task4():  
 def print\_string\_list(array):  
 for i in range(len(array)):  
 print(array[i])  
  
 def get\_string\_array\_from\_console(n):  
 console\_array = []  
 for i in range(n):  
 console\_array.append(str(input('Введите математический термин в ед. числе ')))  
 return console\_array  
  
 def proverka(array1, array2):  
 for i in range(len(array1)):  
 if array1[i] in array2:  
 array1[i] += ' ✓'  
 return array1  
  
 math\_termins = ('Вектор', 'Предел', 'Интеграл', 'Производная', 'График функции', 'Матрица',  
 'Определитель', 'Тригонометрическая функция', 'Гиперболическая функция', 'Асимптота',  
 'Точка экстремума', 'Точка перегиба')  
  
 user\_array = proverka(get\_string\_array\_from\_console(len(math\_termins)), math\_termins)  
 print\_string\_list(user\_array)

## 4.4 Тестирование работы программы с проверкой

|  |  |
| --- | --- |
| Ввод в консоль | Решение Python |
|  |  |

# Задание 5

## Постановка задачи

## Написать функцию task5(), реализующую следующий функционал:

## 1. Выводится сообщение: Введите текст:

## 2. Выполняется цикл. На каждом шаге цикла (всего n шагов) необходимо получить строку текста и записать её в файл.

## 5.2 Алгоритм решения

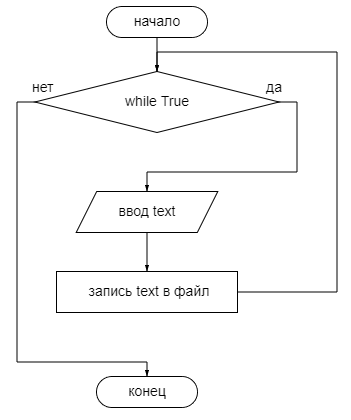


Рис.9 -Блок-схема программы №5

## 5.3 Решение задачи, код программы

def task5():  
 with open('task5.txt', 'a') as file:  
 print('Введите текст ')  
 while True:  
 text = str(input())  
 file.write(text + '\n')

## 5.4 Тестирование работы программы с проверкой

|  |  |
| --- | --- |
|  |  |

В результате работы программ введённый текст был записан в файл task5.txt

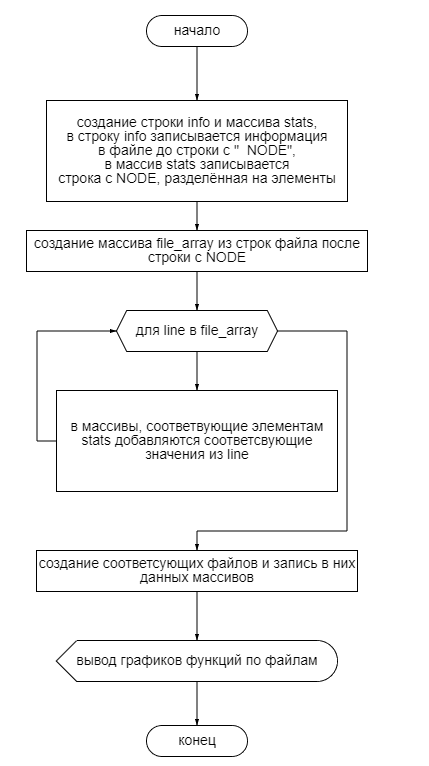
# Задание 6

## Постановка задачи

Необходимо добавить в проект приложенный файл variant2.txt, содержащий результаты численного эксперимента. Написать функцию task6(), реализующую следующий функционал:

1. Производится чтение и анализ файла. Считываются названия(характеристики) столбцов данных (в строке, начинающейся с ключевого слова NODE).
2. Создаются соответствующие характеристикам одномерные массивы данных и получаются их значения из файла.
3. Создаются файлы характеристик с расширением .dat и в каждый файл записываются значения соответствующего массива.
4. Выводятся в виде графиков значения массивов данных.

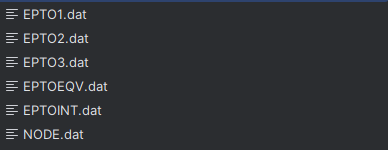
## Алгоритм решения

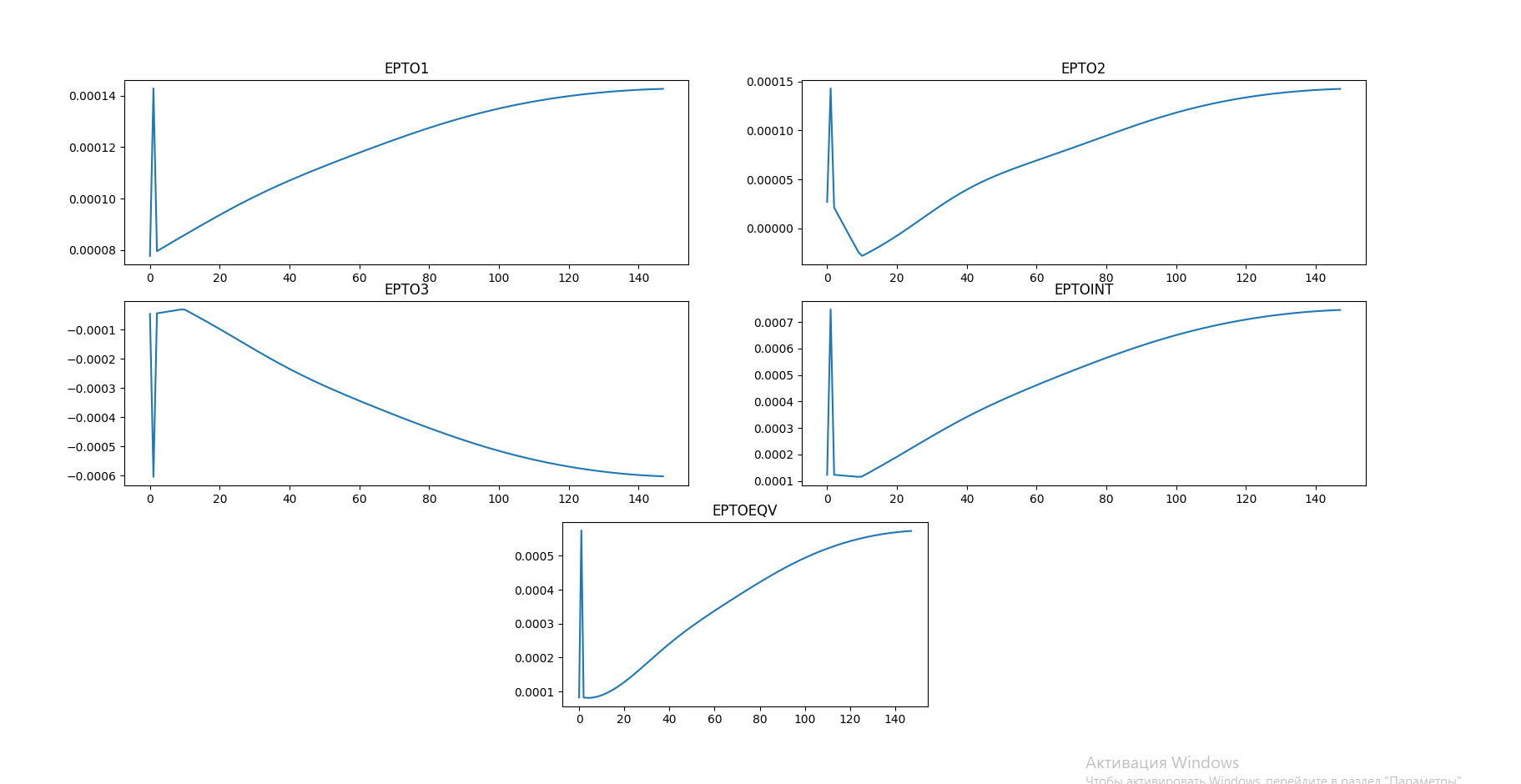
  
Рис.10 – Блок-схема программы №6

## Решение задачи, код программы

1. def task6():  
    with open('variant5.txt', 'r') as file:  
     
    info = ''  
    while 1:  
    line = file.readline()  
    if ' NODE' not in line:  
    info += line  
    else:  
    stats = line.split()  
    break  
     
    file\_array = [line for line in file.readlines()]  
    node, epto1, epto2, epto3, eptoint, eptoeqv = [], [], [], [], [], []  
    for line in file\_array:  
    node.append(line[4:8])  
    epto1.append(line[9:22])  
    epto2.append(line[22:35])  
    epto3.append(line[35:48])  
    eptoint.append(line[48:61])  
    eptoeqv.append(line[61:74])  
     
    array = (node, epto1, epto2, epto3, eptoint, eptoeqv)  
    for i in range(len(array)):  
    file = open(f'{stats[i]}.dat', 'w')  
    for el in array[i]:  
    (file.write(el + '\n'))  
     
    axes1, axes2 = plt.subplot(321), plt.subplot(322)  
    axes3, axes4 = plt.subplot(323), plt.subplot(324)  
    axes5 = plt.subplot(338)  
    with open('EPTO1.dat', 'r') as file:  
    y = [float(a) for a in file]  
    x = [a for a in range(len(y))]  
    axes1.plot(x, y)  
    axes1.set\_title('EPTO1')  
    with open('EPTO2.dat', 'r') as file:  
    y = [float(a) for a in file]  
    x = [a for a in range(len(y))]  
    axes2.set\_title('EPTO2')  
    axes2.plot(x, y)  
    with open('EPTO3.dat', 'r') as file:  
    y = [float(a) for a in file]  
    x = [a for a in range(len(y))]  
    axes3.set\_title('EPTO3')  
    axes3.plot(x, y)  
    with open('EPTOINT.dat', 'r') as file:  
    y = [float(a) for a in file]  
    x = [a for a in range(len(y))]  
    axes4.set\_title('EPTOINT')  
    axes4.plot(x, y)  
    with open('EPTOEQV.dat', 'r') as file:  
    y = [float(a) for a in file]  
    x = [a for a in range(len(y))]  
    axes5.set\_title('EPTOEQV')  
    axes5.plot(x, y)  
    plt.show()

## 6.4 Тестирование работы программы с проверкой





В результате работы программы были созданы 6 файлов с данными, и по ним были построены графики.