Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №4

за 1 семестр

По дисциплине: «Языки программирования»

Тема: «Основы языка Python»

Выполнил:

Студент 2 курса

Группы ПО-4(1)

Иваненко И. Л.

Проверил:

Хацкевич М. В.

2020

Лабораторная работа №4

Основы языка Python

Цель работы: ознакомиться с основами языка Python.

Структура проекта:

Задание 1 Вариант 10(2):

1. Напишите программу для решения примера по варианту. Предусмотрите проверку деления на ноль. Все необходимые переменные пользователь вводит через консоль.



2. Дан произвольный список, содержащий и строки, и числа.



3. Дан произвольный список, содержащий только числа



4. Дан произвольный список, содержащий только числа



Задание 2 Вариант 10(2)

1. Пусть задано некоторое число my\_number. Пользователь вводит с клавиатуры свое число user\_number.



2. Пусть задан список, содержащий строки.



3. Сгенерируйте и выведите:



4. Пусть дана строка:



Задание 3 Вариант 10(2):

Определить структуру для хранения матрицы.

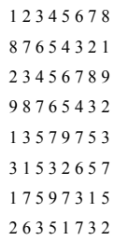
1. Напишите функцию сложения по строкам

2. Напишите функцию сложения по столбцам.

3. Напишите функцию сложения по строкам четных элементов.

4. Напишите функцию сложения по стобцам четных элементов.

Дана матрица:



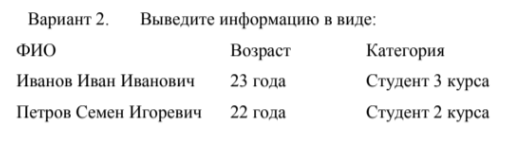
Задание 4 Вариант 10(2):

1. Пусть дана строка, состоящая из слов, пробелов и знаков препинания. На основании этой строки создайте новую и выведите:

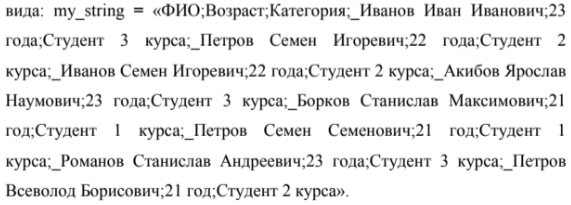
Вариант 2: Содержащую только слова, в которых первые две буквы – “Ли”

2. Пусть дана строковая переменная, содержащая информацию о студентах:

my\_string = “Ф;И;O;Возраст;Категория;\_Иванов;Иван;Иванович;23 года: Студент 3 курса;\_Петров;Семен;Игоревич;22 года;Студент 2 курса”



3. Пусть дана строковая переменная, содержащая информацию о студентах вида:





4. Дана строка произвольной длины. Выведите информацию о том, сколько в ней символов и слов.

Задание 5 Вариант 10(2)

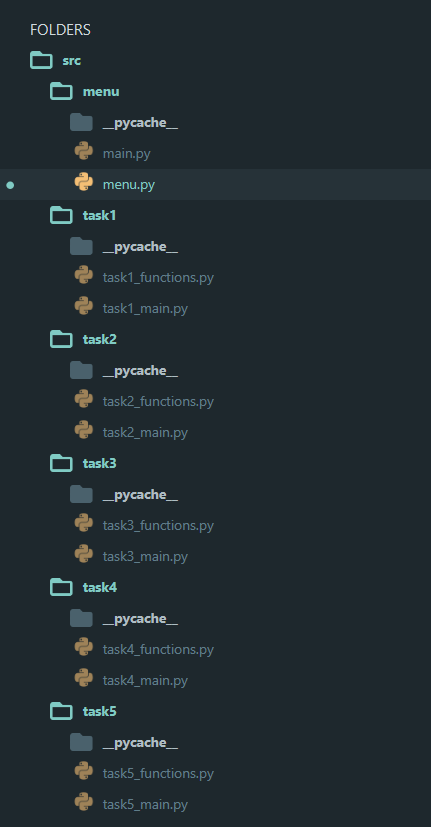
1. Дана матрица размером NxN. Представить матрицу в виде списка

2. Дан список из 10 элементов. Удалить все четные элементы и добавить 2 новых

3. Вывести список студентов определенной группы в определенном формате

4. Вывести студентов, чья фамилия меньше 7 букв

Структура Проекта:



**Модуль task1\_functions:**

# Функция выводит решенный пример

def print\_expression\_result(a, b, c, d, k):

if (b != 0 and a != 0):

print("Результат вычисления выражения:")

print( abs(((a\*\*2 - b\*\*3 - c\*\*3 \* a\*\*2 ) \* (b - c + c\*(k - d/b\*\*3)) - (k/b - k/a)\*c)\*\*2 - 20000) )

else:

print("На ноль делить нельзя")

#Функция выводит нечетные элементы списка

def print\_odd\_elements(lst):

for element in lst:

if (type(element) is float or type(element) is int) and element % 2 != 0:

print(element)

#Функция возвращает результат сложения всех чисел списка от 1 до 10

def get\_calculate\_result(lst):

calculate\_result = 0

for element in lst:

if( element >= 1 and element <= 10):

calculate\_result += element

return calculate\_result

**Модуль task2\_functions:**

#Функция запрашивает user\_number до тех пор, пока оно не будет равно my\_number

def ask\_user\_number(my\_number):

while True:

user\_number = int(input("Введите user\_number: "))

if user\_number == my\_number:

print("user\_number совпал c my\_number!!")

break

#Функция возвращает построчно выводит все строки размером менее 10 символов из списка

def get\_list\_line(lst):

for line in lst:

if len(line) < 10:

print(line)

#Функция возвращает строку размером N, состоящую из букв R

def get\_R\_line(N):

R\_line = ""

for i in range(N):

R\_line += "R"

return R\_line

#Функция возвращает строку без чисел:

def remove\_digits(string):

new\_string = ""

for symbol in string:

if symbol.isdigit():

continue

else:

new\_string += symbol

return new\_string

**Модуль task3\_functions:**

#Функция вывода матрицы

def print\_matrix(matrix):

for row in matrix:

print(row)

#Функция сложения по строкам

def rows\_sum(matrix):

for row in matrix:

row\_sum = 0

for element in row:

row\_sum += element

print(str(row) + " == " + str(row\_sum))

#Функция сложения по столбцам

def cols\_sum(matrix):

col\_sum = []

for i in range(len(matrix[0])):

col\_sum.append(0)

for i in range(len(matrix)):

for j in range(len(matrix[i])):

col\_sum[j] += matrix[i][j]

print\_matrix(matrix)

print("========================")

print(col\_sum)

#Функция сложения по строкам четных элементов.

def even\_rows\_sum(matrix):

for row in matrix:

row\_sum = 0

for element in row:

if element % 2 == 0:

row\_sum += element

print(str(row) + " == " + str(row\_sum))

#Функция сложения по столбцам четных элементов

def even\_cols\_sum(matrix):

col\_sum = []

for i in range(len(matrix[0])):

col\_sum.append(0)

for i in range(len(matrix)):

for j in range(len(matrix[i])):

if matrix[i][j] % 2 == 0:

col\_sum[j] += matrix[i][j]

print\_matrix(matrix)

print("========================")

print(col\_sum)

**Модуль task4\_functions:**

#Функция возвращает строку, состоящую слова, в которых первые 2 буквы - Ли

def get\_str(string):

string = string.replace(",", "")

string = string.replace(".", "")

string = string.replace("!", "")

string = string.replace("?", "")

string = string.replace(";", "")

string = string.replace(";", "-")

string = string.split()

new\_string = ""

for word in string:

if word[0] == "Л" and word[1] == "и":

new\_string = new\_string + word + " "

return new\_string

#Функция возврощает массив, содержащий информацию о студентах

def get\_str\_matrix(string, cols):

string = string.replace("\_", "")

string = string.split(";")

str\_matrix = []

str\_matrix\_row = []

for i in range(len(string)):

if i == len(string) - 1:

str\_matrix\_row.append(string[i])

str\_matrix.append(str\_matrix\_row)

break

if i != 0 and i % cols == 0:

str\_matrix.append(str\_matrix\_row)

str\_matrix\_row = []

str\_matrix\_row.append(string[i])

return str\_matrix

#Функция форматированного вывода

def get\_format\_str(string):

str\_matrix = get\_str\_matrix(string, 5)

for i in range(len(str\_matrix)):

if i == 0:

print("%-25s %-15s %-15s" % (str\_matrix[i][0] + str\_matrix[i][1] + str\_matrix[i][2],

str\_matrix[i][3],

str\_matrix[i][4],

))

else:

print("%-25s %-15s %-15s" % (str\_matrix[i][0] + " " + str\_matrix[i][1] + " " + str\_matrix[i][2],

str\_matrix[i][3],

str\_matrix[i][4],

))

def get\_students(string):

str\_matrix = get\_str\_matrix(string, 3)

for student in str\_matrix:

if student[1][0] == "2" and student[1][1] == "1":

print(student)

def calculate\_data(string):

data\_list = string.split()

amount\_of\_symbols = 0

amount\_of\_words = 0

for word in data\_list:

amount\_of\_words += 1

amount\_of\_symbols += len(word)

print("Количество слов: " + str(amount\_of\_words))

print("Количество символов: " + str(amount\_of\_symbols))