МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

**Разработка программных систем**

Знакомство с языком программирования Python

Вариант 3

Выполнил студент группы ИВТ-31 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Кудяшев Я.Ю./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Чистяков Г.А./

Киров 2022

1. Задание

В ходе выполнения лабораторной работы необходимо решить следующие задачи:

* Установить на компьютер интерпретатор языка Python;
* Установить на компьютер подходящую среду разработки;
* Настроить рабочее окружение;
* Реализовать в отдельном модуле группу методов;
* Продемонстрировать работу модуля на подготовленном сценарии;

Модуль должен иметь 4 метода:

1. def number\_of\_words(line) – возвращает число слов в заданной строке;
2. def number\_of\_symbols(line) – возвращает число символов в заданной строке;
3. def number\_of\_universal\_symbols(line) – возвращает число уникальных символов в заданной строке;
4. def number\_of\_defined\_characters(line,symbol) – возвращает число вхождений заданного символа в строке.

1. Листинг программы

Листинг программы приведен в приложении А.

1. Вывод

В ходе выполнения лабораторной работы были изучены основные конструкции языка программирования Python, структура программы, стандартные средства ввода/вывода; изучен основной функционал интегрированной среды разработки Visual Studio для работы с Python; написан модуль для анализа строк. Кроме этого, для реализации методов был применен функциональный стиль написания.

Приложение А

(обязательное)

Листинг программы

def number\_of\_words(line):

return sum(map(lambda x : int(x.isalpha()), line.split()))

def number\_of\_defined\_characters(line,symbol):

return line.count(symbol)

def number\_of\_symbols(line):

return len(line)

def number\_of\_universal\_symbols(line):

return len(set(line))

def menu():

line = input("Enter the string: ")

print("1. Number of symbols")

print("2. Number of words")

print("3. Number of defined characters")

print("4. Number of universal symbols")

print("5. Exit")

what\_to\_do = input("Select an action: ")

if what\_to\_do == "1":

count = number\_of\_symbols(line)

print("Number of symbols is " + str(count))

menu()

elif what\_to\_do == "2":

count = number\_of\_words(line)

print("Number of words is " + str(count))

menu()

elif what\_to\_do == "3":

symbol = input("Enter a symbol to search for: ")

count = number\_of\_defined\_characters(line, symbol)

print("There are " + str(count) + ' symbols "' + symbol + '"')

menu()

elif what\_to\_do == "4":

count = number\_of\_universal\_symbols(line)

print("Number of universal symbols are " + str(count))

menu()

elif what\_to\_do == "5": exit

else:

print("\nWrong number. Try again.")

menu()