МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ

Отчет о лабораторной работе №14 по дисциплине «Основы программной инженерии»

Выполнил:

Мамонтов Д.В.,

2 курс, группа ПИЖ-б-о-20-1,

Проверил:

Доцент кафедры инфокоммуникаций,

Воронкин Р.А.

Ставрополь, 2022 г

**ХОД РАБОТЫ**

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 – использование локальной переменной

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 – переменная области видимости enclosing

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 – использование глобальных переменных

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4 – использование замыканий

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5 – использование функции nonlocal

**ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ**

Задание:

Используя замыкания функций, объявите внутреннюю функцию, которая преобразует строку из списка целых чисел, записанных через пробел, либо в список, либо в кортеж. Тип коллекции определяется параметром type внешней функции. Если type = 'list' , то используется список, иначе – кортеж. Далее, на вход программы поступает две строки: первая – это значение для параметра type; вторая – список целых чисел, записанных через пробел. С помощью реализованного замыкания преобразовать эту строку в  
соответствующую коллекцию. Результат работы замыкания выведите на экран

Код:

def func(func\_type: str):  
 def inner(value):  
 gen = (e for e in value.split())  
 if func\_type == 'list':  
 return list(gen)  
 return tuple(gen)  
  
 return inner  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 x = input()  
 y = input()  
 print(func(x)(y))

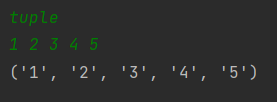


Рисунок 1 – результат работы программы при коллекции tuple

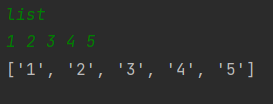


Рисунок 2 – результат работы программы при коллекции list

**КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Что такое замыкание?

Замыкание — это функция, в теле которой присутствуют ссылки на переменные, объявленные вне тела этой функции в окружающем коде и не являющиеся ее параметрами.

1. Как реализованы замыкания в языке программирования Python?

• У нас должна быть вложенная функция (функция внутри функции).

• Вложенная функция должна ссылаться на значение, определенное в объемлющей функции.

• Объемлющая функция должна возвращать вложенную функцию.

1. Что подразумевает под собой область видимости Local?

Эту область видимости имеют переменные, которые создаются и используются внутри функций.

1. Что подразумевает под собой область видимости Enclosing?

Суть данной области видимости в том, что внутри функции могут быть вложенные функции и локальные переменные, так вот локальная переменная функции для ее вложенной функции находится в enclosing области видимости.

1. Что подразумевает под собой область видимости Global?

Переменные области видимости global – это глобальные переменные уровня модуля (модуль – это файл с расширением .py).

1. Что подразумевает под собой область видимости Build-in?

Уровень Python интерпретатора. В рамках этой области видимости находятся функции open, len и т. п., также туда входят исключения.

1. Как использовать замыкания в языке программирования Python?

def mul(a):

def helper(b):

return a \* b

return helper

1. Как замыкания могут быть использованы для построения иерархических данных?

В общем случае, операция комбинирования объектов данных обладает свойством замыкания в том случае, если результаты соединения объектов с помощью этой операции сами могут соединяться этой же операцией