

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

**Отчет по лабораторной работе №6
Замыкания в языке Python.**

по дисциплине «Технологии программирования и алгоритмизации»

Выполнила студентка группы ИВТ-б-о-20-1

Новикова В.С. « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____

(подпись)

Ставрополь 2021

Цель работы: приобретение навыков по работе с замыканиями при написании программ с помощью языка программирования Python версии 3х.

Ход работы:

Ссылка на репозиторий: https://github.com/Valentina1502/LABA_6

Пример 1. (рис. 1).

Пример использования замыканий.

В функции `fun1()` объявлена локальная переменная `x`, значение которой определяется аргументом `a`. В функции `fun2()` используются эта же переменная `x`, `nonlocal x` указывает на то, что эта переменная не является локальной, следовательно, ее значение будет взято из ближайшей области видимости, в которой существует переменная с таким же именем. В нашем случае – это область `enclosing`, в которой этой переменной `x` присваивается значение `a * 3`. Также как и в предыдущем случае, на переменную `x` после вызова `fun1(4)`, сохраняется ссылка, поэтому она не уничтожается.

Код:

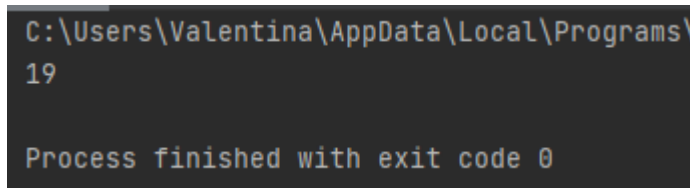
```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def fun1(a):
    x = a * 3

    def fun2(b):
        nonlocal x
        return b + x

    return fun2

if __name__ == '__main__':
    test = fun1(4)
    print(test)
    print(test(7))
```



```
C:\Users\Valentina\AppData\Local\Programs\
19

Process finished with exit code 0
```

Рисунок 1 – Пример 1

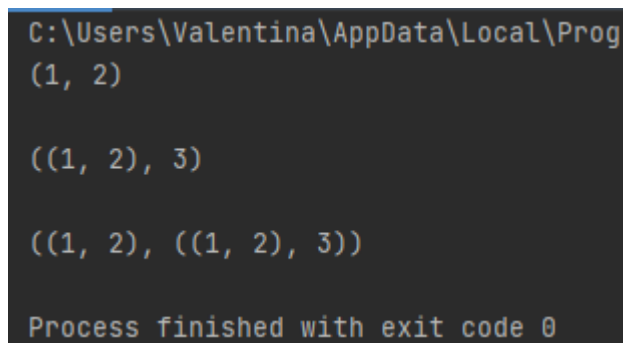
Задание 1 (рис. 2-3):

Свойства замыканий – можно строить иерархические структуры данных. Создадим функцию `tpl()`, которая на вход принимает два аргумента и возвращает кортеж. Эта функция реализует операцию «объединения элементов в кортеж»

Код:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == '__main__':
    tpl = lambda a, b: (a, b)
    p1 = tpl(1, 2)
    print(f'{p1} \n ')
    p2 = tpl(p1, 3)
    print(f'{p2} \n')
    p3 = tpl(p1, p2)
    print(f'{p3} ')
```



```
C:\Users\Valentina\AppData\Local\Prog
(1, 2)

((1, 2), 3)

((1, 2), ((1, 2), 3))

Process finished with exit code 0
```

Рисунок 2 – Пример 2

Задание 1. Вариант 11 (1) (рис. 3):

Используя замыкания функций, определите вложенную функцию, которая бы увеличивала значение переданного параметра на 3 и возвращала бы вычисленный результат. Вызовите внешнюю функцию для получения ссылки на внутреннюю функцию и присвойте ее переменной с именем `cnt`. Затем, вызовите внутреннюю функцию через переменную `cnt` со значением `k`, введенным с клавиатуры.

Код:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def f1():
    def f2(p):
        result = p + 3
        return result
    return f2

if __name__ == '__main__':
    cnt = f1()
    k = int(input('Введите число \n'))
    print(f'Результат вычислений: {cnt(k)}')
```

```
C:\Users\Valentina\AppData\Local\Programs\Python\Python39\python.exe
Введите число
156
Результат вычислений: 159
Process finished with exit code 0
```

Рисунок 3 – Задание 1

Контрольные вопросы:

1. Что такое замыкание?

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

2. Как реализованы замыкания в языке программирования Python?

В Python замыкания реализованы путём вложенных функций, где аргумент отправляется только во внешнюю функцию, а внутренняя функция, несмотря на отсутствие параметров успешно выполняет операции с этим аргументом.

3. Что подразумевает под собой область видимости Local?

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

4. 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.

5. Что подразумевает под собой область видимости Global?

Переменные области видимости global – это глобальные переменные уровня модуля

6. 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. Эти

сущности доступны в любом модуле Python и не требуют предварительного импорта. Built-in – это максимально широкая область видимости.

7. Как использовать замыкания в языке программирования Python?

Рассмотрим на примере:

```
>>> def mul(a):  
    def helper(b):  
        return a * b  
    return helper  
>>> mul(5)(2)
```

Программа перемножает переданные в неё числа путём замыкания в функции.

8. 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.

Вывод: при выполнении заданий были приобретены навыки по работе с замыканиями при написании программ с помощью языка программирования Python.