

**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 №2.4 по дисциплине основы
программной инженерии**

Выполнил:

Кожухов Филипп Денисович,
2 курс, группа ПИЖ-б-о-20-1,

Проверил:

Доцент кафедры
прикладной математики и
компьютерной
безопасности, Воронкин Р.А.

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2021 г.

ВЫПОЛНЕНИЕ:

```
1  ▶  #!/usr/bin/env python3
2      # -*- coding: utf-8 -*-
3
4
5      def higher_order(func):
6          print('Получена функция {} в качестве аргумента'.format(func))
7          func()
8          return func
9
10
11     def wrapper_function():
12         def hello_world():
13             print('Hello world!')
14         hello_world()
15
16
17     def hello_world():
18         print('Hello world!')
19
20
21     ▶  if __name__ == "__main__":
22         print(type(hello_world))
23
24
25         class Hello:
26
27             pass
28
29
30         print(type(Hello))
31         print(type(10))
32         hello = hello_world
33         hello()
34         wrapper_function()
35     ▶  print(higher_order(hello_world))
```

Рисунок 1 - Код примера №1

```
C:\Python37\python.exe C:/Users/student-09-525/PycharmProjects/lab_15/ex1.py
<class 'function'>
<class 'type'>
<class 'int'>
Hello world!
Hello world!
Получена функция <function hello_world at 0x0285D8A0> в качестве аргумента
Hello world!
<function hello_world at 0x0285D8A0>

Process finished with exit code 0
```

Рисунок 2 - Вывод программы

```

1  ▶  #!/usr/bin/env python3
2      #- coding: utf-8 -*-
3
4
5      def decorator_function(func):
6          def wrapper():
7              print('Функция-обёртка!')
8              print('Оборачиваемая функция: {}'.format(func))
9              print('Выполняем обёрнутую функцию...')
10             func()
11             print('Выходим из обёртки')
12         return wrapper
13
14
15     @decorator_function
16     def hello_world():
17         print('Hello world!')
18
19
20  ▶  if __name__ == "__main__":
21     🟡 hello_world()
22

```

Рисунок 3 - Код примера №2

```

C:\Python37\python.exe C:/Users/student-09-525/PycharmProjects/lab_15/ex2.py
Функция-обёртка!
Оборачиваемая функция: <function hello_world at 0x00F9D8A0>
Выполняем обёрнутую функцию...
Hello world!
Выходим из обёртки

Process finished with exit code 0

```

Рисунок 4 - Вывод программы

```

1 ▶  #!/usr/bin/env python3
2    # -*- coding: utf-8 -*-
3
4
5    def benchmark(func):
6        import time
7
8        def wrapper():
9            start = time.time()
10           func()
11           end = time.time()
12           print('[*] Время выполнения: {} секунд.'.format(end - start))
13       return wrapper
14
15
16   @benchmark
17   def fetch_webpage():
18       import requests
19       webpage = requests.get('https://www.google.com')
20
21
22 ▶  if __name__ == "__main__":
23       fetch_webpage()
24

```

Рисунок 5 - Код примера №3

```

C:\Python37\python.exe C:/Users/student-09-525/PycharmProjects/lab_15/ex3.py
[*] Время выполнения: 1.1455731391906738 секунд.

Process finished with exit code 0

```

Рисунок 6 - Вывод программы

```

1  ▶  #!/usr/bin/env python3
2      # -*- coding: utf-8 -*-
3
4
5      def benchmark(func):
6          import time
7
8          def wrapper(*args, **kwargs):
9              start = time.time()
10             return_value = func(*args, **kwargs)
11             end = time.time()
12             print('[*] Время выполнения: {} секунд.'.format(end - start))
13             return return_value
14
15         return wrapper
16
17
18     @benchmark
19     def fetch_webpage(url):
20         import requests
21         webpage = requests.get(url)
22         return webpage.text
23
24
25  ▶  if __name__ == "__main__":
26      webpage = fetch_webpage('https://google.com')
27      print(webpage)
28

```

Рисунок 7 - Код примера №4

```

C:\Python37\python.exe C:/Users/student-09-525/PycharmProjects/lab_15/ex4.py
[*] Время выполнения: 0.7344775199890137 секунд.
<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="ru"><head><meta content="#1055;&#1086;&#1080;&#1089;&#1
var f=this||self;var h,k=[];function l(a){for(var b;a&&(!a.getAttribute||!(b=a.getAttribute("eid"))));)a=a.parentNode;return b||h}func
function n(a,b,c,d,g){var e="";c||-1===b.search("&ei=")|| (e="&ei="+l(d),-1===b.search("&lei=")&&(d=m(d))&&(e+="&lei="+d));d="";!c&&f.
google.y={};google.sy=[];google.x=function(a,b){if(a)var c=a.id;else{do c=Math.random();while(google.y[c])}google.y[c]=[a,b];return!1
document.documentElement.addEventListener("submit",function(b){var a;if(a=b.target){var c=a.getAttribute("data-submitfalse");a="1"===
</style><style>body,td,a,p,.h{font-family:arial,sans-serif}body{margin:0;overflow-y:scroll}#gog{padding:3px 8px 0}td{line-height:.8em
var f=this||self;var g,h=null≠(g=f.mei)?g:1,m,n=null≠(m=f.sdo)?m:!0,p=0,q,r=google.erd,u=r.jsr;google.ml=function(a,b,d,k,c){c=void
e);var l=a.fileName;l&&(b+="&script="+c(l),e&&l===window.location.href&&(e=document.documentElement.outerHTML.split("\n")[e],b+="&cad
if (!iesg){document.f&&document.f.q.focus();document.gbqf&&document.gbqf.q.focus();}
}
})();</script><div id="mngb"><div id=gbar><nobr><b class=gb1&#1055;&#1086;&#1080;&#1089;&#1082;</b> <a class=gb1 href="https://www.i
else top.location='/doodles/';:}}):</script><input value="AHkkrS4AAAAAYiMtJIV0Pz7-l11vMJ0ooiLEEOCqQVZU" name="iflsiq" ttype="hidden

```

Рисунок 8 - Вывод программы

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

Вариант 5

```
1 ▶ #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def start_sum(start):
5     def wrapper(func):
6         def wrapped(*args):
7             print(f"Entering the wrapper with start value of {start}")
8             print(f"Processing wrapped function: {func}")
9             string = args[0]
10            lst = string.split()
11            lst = lst[start:]
12            start_string = " ".join(lst)
13            return_sum = func(start_string)
14            print("Returning from wrapper")
15            return return_sum
16        return wrapper
17    return decorator
18
19
20 #@start_sum(start=5)
21 def str_summary(string):
22     a = [int(s) for s in string.split()]
23     summary = sum(elem for elem in a)
24     return summary
25
26
27 ▶ if __name__ == "__main__":
28     string_inp = input("Enter the str numbers: ")
29     print(f"The sum is: {str_summary(string_inp)}")
30
```

Рисунок 9 - Код ИДЗ

```
C:\Python37\python.exe C:/Users/student-09-525/PycharmProjects/lab_15/individual.py
Enter the str numbers: 1 24 5
The sum is: 30

Process finished with exit code 0
|
```

Рисунок 10 - Вывод программы

ОТВЕТЫ НА ВОПРОСЫ

1. Что такое декоратор?

Декоратор — это функция, которая позволяет обернуть другую функцию для расширения её функциональности без непосредственного изменения её кода.

2. Почему функции являются объектами первого класса?

Потому что с ними можно делать всё то же, что и с любым другим объектом: передавать как параметр, возвращать из функции и присваивать переменной.

3. Каково назначение функций высших порядков?

Функции высших порядков — это такие функции, которые могут принимать в качестве аргументов и возвращать другие функции. Они предназначены для передачи функций как параметры другим функциям, а также возвращать их как значения

4. Как работают декораторы?

Декоратор — это функция, которая позволяет обернуть другую функцию для расширения её функциональности без непосредственного изменения её кода.

5. Какова структура декоратора функций?

Объявляется декоратор, который принимает функцию как аргумент, внутри декоратора объявляется обёртка, которая выполняется функцию с изменёнными условиями.

6. Самостоятельно изучить как можно передать параметры декоратору, а не декорируемой функции?

Поместить декоратор внутри функции, которая будет принимать некоторые параметры, а затем использовать эти параметры в декораторе.