

ГУАП

КАФЕДРА № 41

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

ассистент

должность, уч. степень, звание

подпись, дата

Е.К. Григорьев

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 1

РАБОТА СО СТРОКАМИ, СПИСКАМИ, КОРТЕЖАМИ

по курсу: ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

4116

подпись, дата

Четвергов В.Ю.

инициалы, фамилия

Санкт-Петербург 2023

Цель работы: познакомиться с основными способами работы со строками, списками и кортежами средствами языка программирования Python. Вариант 7

### Часть 1. Задания по строкам

7. Даны две строки str1 и str2. Программа должна создать новую строку str3 путем добавления str2 в середину str1, за счет использования срезов. Полученный результат выведите в терминал. Например: str1 = «Мама», str2 = «Раму» -> «МаРамума».

```
str1 = input()

while (len(str1) % 2 > 0) or (len(str1) % 2 < 0):

    print('число символов в первой строке нечётное')

    str1 = input()

str2 = input()

a = len(str1)

b = int(a/2)

str3 = str1[:b]+str2+str1[b:]

print(str3)
```

9. Даны две строки str1 и str2. Программа должна создать новую строку str3 состоящую из первого, среднего и последнего символов строк str1 и str2. Полученный результат выведите в терминал. Например: str1 = «Мама», str2 = «Утром» -> «МУмрам».

```
str1 = input()

str2 = input()

MED1 = len(str1)-1

MED2 = len(str2)-1

a = int(round(MED1/2))

b = int(round(MED2/2))

str3 = str1[:1]+str2[:1]+str1[a]+str2[b]+str1[MED1]+str2[MED2]

print(str3)
```

11. Дана строка «В данной главе мы ознакомились с тем, как выполнить установку интерпретатора». Используя пробелы, разбейте ее на части, сформировав список. Полученный результат выведите в терминал.

```
line = "В данной главе мы ознакомились с тем, как выполнить установку  
интерпретатора"
```

```
a = line.split()
```

```
for x in range(len(a)):
```

```
    print(a[x])
```

## **Часть 2. Задания по спискам**

2. Дан список `my_list = [1, 2, 3, 4, 5, 6]`. Добавьте в его конец значение 2 и выведите в терминал полученный результат.

```
my_list = [1, 2, 3, 4, 5, 6]
```

```
my_list.append(2)
```

```
print(my_list)
```

4. Дан список `my_list = [1, 2, 3, 10]`. Вставьте новый элемент со значением 12 на 2-ю позицию и выведите в терминал полученный результат

```
my_list = [1, 2, 3, 10]
```

```
print(my_list)
```

```
my_list.insert(2, 12)
```

```
print(my_list)
```

17. Дан список `my_list = [1, 2, 3, 4, 5, 6, 2, 3, 2, 1, 2, 2, 5]`. Не используя циклов отсортируйте его элементы по возрастанию. Полученный результат выведите в терминал

```
my_list = [1, 2, 3, 4, 5, 6, 2, 3, 2, 1, 2, 2, 5]
```

```
print(my_list)
```

```
my_list.sort()
```

```
print(my_list)
```

19. Дан список `my_list = [1, 2, 3, 4, 5, 1, 2, 'a', 'b', 2, 5]`. Получите номер индекса элемента со значением 'a' и выведите в терминал полученный результат. Для решения задачи запрещено использовать циклы

```
my_list = [1, 2, 3, 4, 5, 1, 2, 'a', 'b', 2, 5]

print(my_list.index('a'))
```

### Часть 3. Задания по кортежам

3. Дан кортеж `my_tuple = (1, 2, 5, 6)`. Измените значение его элемента 5 на 'b' и выведите в терминал полученный результат.

```
my_tuple = (1, 2, 5, 6)

lst = list(my_tuple)

lst[2] = 'b'

t = tuple(lst)

print(t)
```

4. Дан кортеж `my_tuple = (1, 2, 5, 6)`. Добавьте новый элемент со значением 12 на 2-ю позицию и выведите в терминал полученный результат

```
my_tuple = (1, 2, 5, 6)

lst = list(my_tuple)

lst.insert(2, 12)

t = tuple(lst)

print(t)
```

6. Дан кортеж `my_tuple = (1, 2, 3, 1, 2, 3, 4, 5, 6, 2, 3, 2, 1, 2)`. Посчитайте количество вхождения в него элементов со значением 3 и выведите в терминал полученный результат. Для решения задачи запрещено использовать циклы

```
my_tuple = (1, 2, 3, 1, 2, 3, 4, 5, 6, 2, 3, 2, 1, 2)

print(my_tuple.count(3))
```

9. Дан кортеж `my_tuple = (0, -2, 81, 42, 6, -6, 23)`. Проверьте входит ли в него элемент со значением `-6` и выведите в терминал полученный результат. Для решения задачи запрещено использовать циклы.

```
my_tuple = (0, -2, 81, 42, 6, -6, 23)
```

```
print(-6 in my_tuple)
```

Выводы работы: в процессе работы ознакомлен со строками, списками и кортежами языка Python