

Практическое занятие №12.

Тема: Составление программ в функциональном стиле в IDE PyCharm Community.

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со списками в IDE PyCharm Community.

Постановка задач №1:

1. Даны две последовательности. Найти элементы, различные для двух последовательностей и их среднее арифметическое.

Постановка задачи №2:

2. Из заданной строки отобразить только цифры. Использовать библиотеку `string`. Строка - `TheGreatPyramidofKhufuatGizawasbuiltabout 2700 BC, 755 feet (230metres) longand 481 feet (147 metres) high.`

Тип алгоритма: Функциональный с использованием генераторов, итераторов, списковых включений.

Текст программы №1:

#Импортирование библиотеки `random`.

```
import random
```

#####ВТОРОЕ РЕШЕНИЕ#####

```
list_number_one = [random.randint(-10, 20) for _ in range(random.randint(2, 8))] #Генерация  
уникального множества
```

```
list_number_two = [random.randint(-10, 20) for _ in range(random.randint(2, 8))] #Генерация  
уникального множества
```

```
list_test = list(filter(lambda x: x not in list_number_two, list_number_one)) #Сортировка элем.  
которые входят только в 1
```

```
list_test_2 = list(filter(lambda x: x not in list_number_one, list_number_two)) #Сортировка элем.  
которые входят только в 2
```

```
#Вывод содержимого множеств
print("Элементы первого множества:", list_number_one)
print("Элементы второго множества:", list_number_two)
print("Итог. Новое множество:", list_test+list_test_2)

# Ответ
print(f"Среднее арифметическое: {sum(list_test+list_test_2) // len(list_test+list_test_2)}")
```

Текст программы №2:

#Импортирование библиотеки string.

```
import string
```

#Задание второе.

```
string_from_work = "TheGreatPyramidofKhufuatGizawasbuiltabout
2700 BC, 755 feet (230metres) longand 481 feet (147 metres) high."
```

#Данная строка по условию

#Проверка каждого элемента в строке.

```
list_digit_from_string = list(filter(lambda x: x in string.digits,
string_from_work)) #Добавляет только True
```

```
print(list_digit_from_string) #Вывод
```

Протокол работы программы №1:

Ввод: генерация random числа

```
list_number_one = [random.randint(-10, 20) for _ in range(random.randint(2, 8))] #Генерация
уникального множества
```

```
list_number_two = [random.randint(-10, 20) for _ in range(random.randint(2, 8))] #Генерация
уникального множества
```

Вывод:

Элементы первого множества: [12, 5, 16]

Элементы второго множества: [9, -5, 6, 17, 7, -6]

Итог. Новое множество: [12, 5, 16, 9, -5, 6, 17, 7, -6]

Среднее арифметическое: 6

Протокол работы программы №2:

Ввод:

#Дана строка

```
string_from_work = "TheGreatPyramidofKhufuatGizawasbuiltabout  
2700 BC, 755 feet (230metres) longand 481 feet (147 metres) high."
```

Вывод:

```
['2', '7', '0', '0', '7', '5', '5', '2', '3', '0', '4', '8', '1', '1', '4', '7']
```

Вывод:

В процессе выполнения практического занятия я выработал навыки составления программ с использованием генераторов, итераторов, списковых включений, генераторов массивов, списков с условиями и без в IDE Pycharm Community. Были использованы языковые конструкции: lambda (анонимной функции), map, list comprehension, filter, zip, reduce, list. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.

