

## Практическое занятие № 7 ЗАДАЧА 1

**Тема:** Составление программ со строками в IDE PyCharm Community

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со строками в IDE PyCharm Community.

**Постановка задачи.**

Даны целые положительные числа N1 и N2 и строки S1 и S2. Получить из этих строк новую строку, содержащую первые N1 символов строки S1 и последние N2 символов строки S2 (в указанном порядке).

**Тип алгоритма:** функциональный.

**Текст программы:**

```
import random
# -*- coding: utf8 -*-
# ПЗ 7 вариант 26 задание 1
# Даны целые положительные числа N1 и N2 и строки S1 и S2. Получить из этих
# строк
# новую строку, содержащую первые N1 символов строки S1 и последние N2
# символов строки S2 (в указанном порядке).

def combine_strings(N1, N2, S1, S2):
    result = S1[:N1] + S2[-N2:]
    return result

N1 = random.randint(1,5)
N2 = random.randint(1,5)
S1 = ''.join([random.choice('abcdf123') for i in range(10)])
S2 = ''.join([random.choice('qwerty456') for i in range(10)])

new_string = combine_strings(N1, N2, S1, S2)

# Выводы
print(f'[INFO] N1 = {N1} | N2 = {N2}')
print(f'[INFO] S1 = {S1} | S2 = {S2}')
print(new_string)
```

**Протокол работы программы:**

[INFO] N1 = 1 | N2 = 1

[INFO] S1 = c11b13abb1 | S2 = r5eq4erer4

c4

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ с функциями в IDE PyCharm Community

В данной программе определена функция `combine_strings`, принимающая целые положительные числа `N1` и `N2`, а также строки `S1` и `S2`. Эта функция формирует новую строку, содержащую первые `N1` символов из строки `S1` и последние `N2` символов из строки `S2` (в указанном порядке).

Для достижения цели использован модуль `random`, который используется для генерации случайных значений.

1. Сначала в функции `combine_strings` создается строка `result` путем объединения первых `N1` символов из строки `S1` и последних `N2` символов из строки `S2`.
2. Далее генерируются случайные значения `N1` и `N2`, а также случайные строки `S1` и `S2`, состоящие из символов `'abcdf123'` и `'qwerty456'` длиной 10 символов соответственно.
3. Затем выводятся сгенерированные случайные значения `N1` и `N2`, а также случайные строки `S1` и `S2`.
4. Наконец, программа выводит новую строку, сформированную с помощью функции `combine_strings`.

Таким образом, при каждом запуске программы будут генерироваться новые случайные значения `N1`, `N2`, `S1` и `S2`, и формироваться новая строка.

Готовые программные коды выложены на [GitHub](#).

## Практическое занятие № 7 ЗАДАЧА 2

**Тема:** Составление программ со строками в IDE PyCharm Community

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со строками в IDE PyCharm Community.

**Постановка задачи.**

Даны строки S и S0. Удалить из строки S все подстроки, совпадающие с S0. Если совпадающих подстрок нет, то вывести строку S без изменений.

**Тип алгоритма:** функциональный.

**Текст программы:**

```
import random
# -*- coding: utf8 -*-
# ПЗ 7 вариант 26 задание 2
# Даны строки S и S0. Удалить из строки S все подстроки, совпадающие с S0.
# Если
# совпадающих подстрок нет, то вывести строку S без изменений.

def remove_substrings(S, S0):
    return S.replace(S0, '')

S = ''.join([random.choice('abc123') if i != 5 else random.choice('ABC') for
i in range(10)])
S0 = ''.join([random.choice('abc123') if i != 5 else random.choice('ABC') for
i in range(2)])
result = remove_substrings(S, S0)

if not S0 in S:
    print(f'[INFO] Подстроки {S0} нет в {S}\n\n{S}')
else:
    print(f'[INFO] Подстрока {S0} найдено в {S} и была удалена из
строки\n\n{result}')
```

**Протокол работы программы:**

[INFO] Подстрока 21 найдено в aba21Ab1ca и была удалена из строки

abaAb1ca

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ с функциями в IDE PyCharm Community

В данной программе определена функция `remove_substrings`, которая удаляет из строки `S` все вхождения подстроки, совпадающей с `S0`. Если совпадений не найдено, программа выводит исходную строку `S`.

Модуль `random` используется для генерации случайных значений.

Взглянув на код, мы видим следующий порядок действий:

1. Определена функция `remove_substrings(S, S0)`, которая использует метод `replace` для удаления всех вхождений подстроки `S0` из строки `S`.
2. Генерируются случайные строки `S` и `S0`, каждая из которых состоит из символов `'abc123'` длиной 10 и 2 символа соответственно.
3. Проверяется, содержит ли строка `S` подстроку `S0`. Если нет, выводится исходная строка `S`.
4. Иначе, выводится информация о том, что подстрока `S0` найдена в строке `S` и была удалена из неё. Затем выводится результат удаления подстроки.

Таким образом, при выполнении программы выводится исходная строка `S` без изменений, если в ней отсутствуют совпадающие подстроки, иначе выводится строка `S` после удаления совпадающих подстрок.

Готовые программные коды выложены на [GitHub](#).