

## 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 с использованием списковых включений, итераторов, генераторов в IDE PyCharm Community

### Постановка задачи.

Вариант 32.

1. В последовательности на  $n$  целых чисел умножить все элементы на первый максимальный элемент.

2. Из заданной строки отобразить только символы пунктуации. Использовать библиотеку string.

Строка: --msg-template="\$FileDir\$\{path}:{line}:{column}:{C}:{symbol}}{msg}"

**Тип алгоритма:** линейный

### Текст программы №1:

```
import random

def multiply_sequence(numbers):
    max_element = max(numbers)

    # Списковое включение
    multiplied_seq = [n * max_element for n in numbers]
    yield from multiplied_seq

# Пример использования функции
sequence = [random.randint(-10, 50) for _ in range(5)]
print(sequence)
result = multiply_sequence(sequence)
result_list = list(result)
print(f'Максимальный элемент: {max(sequence)} \n')

# Вывод результата
for num in result_list:
    print(num)
```

### Протокол работы программы №1:

[6, 2, 16, 35, 0]

Максимальный элемент: 35

210

70

560

1225

0

Process finished with exit code 0

**Текст программы №2:**

```
import string

def get_punctuation_chars(text):
    # Списковое включение
    punctuation_chars = [char for char in text if char in string.punctuation]
    yield from punctuation_chars

my_string = '--msg-
template="$FileDir${path}:{line}:{column}:{C}:{symbol}) {msg}"'
punctuation_chars = get_punctuation_chars(my_string)
print(list(punctuation_chars))
```

**Протокол работы программы №2:**

```
[',', '-', '=', '"', '$', '$', '\\', '{', '}',
':', '{', '}', ':', '{', '}', ':', '{', '}', ':', '(',
'{', '}', ')', '{', '}', '"', '-', '-', '-', '=',
'"', '$', '$', '\\', '{', '}', ':', '{', '}', ':',
'{', '}', ':', '{', '}', ':', '(', '{', '}', ')', '{',
'}', '"', '-', '-', '-', '=', '"', '$', '$', '\\',
'{', '}', ':', '{', '}', ':', '{', '}', ':', '{', '}',
':', '(', '{', '}', ')', '{', '}', '"']
```

Process finished with exit code 0

**Вывод:**

закрепил усвоенные знания, понятия, алгоритмы,

основные принципы составления программ, приобрел навыки составления

программ с использованием списковых включений, итераторов, генераторов в IDE

PyCharm Community