

Практическое занятие № 12

Тема: Составление программ в функциональном стиле.

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 №1.

Организовать и вывести последовательность A из n чисел. Из последовательности A получить две последовательности B и C: в последовательности B – четные элементы A, в C – нечетные элементы A. Произвести суммирование соответствующих элементов последовательностей B и C. Найти минимальный элемент полученной последовательности.

Текст программы:

```
from functools import reduce

def create_sequence(n):
    return list(range(1, n + 1))

# четные элементы последовательности
def get_even_elements(sequence):
    return [x for x in sequence if x % 2 == 0]

# нечетные элементы последовательности
def get_odd_elements(sequence):
    return [x for x in sequence if x % 2 != 0]

# суммирование соответствующих элементов двух последовательностей
def sum_corresponding_elements(sequence1, sequence2):
    return [x + y for x, y in zip(sequence1, sequence2)]

# поиск минимального элемента последовательности
def find_min(sequence):
    return reduce(min, sequence)

sequence = create_sequence(10)

# четные и нечетные элементы последовательности
even_elements = get_even_elements(sequence)
odd_elements = get_odd_elements(sequence)

# сумма соответствующих элементов четной и нечетной последовательностей
summed_elements = sum_corresponding_elements(even_elements, odd_elements)

# минимальный элемент полученной последовательности
min_element = find_min(summed_elements)

print(f"Последовательность A: {sequence}")
print(f"Четные элементы последовательности A: {even_elements}")
print(f"Нечетные элементы последовательности A: {odd_elements}")
print(f"Сумма соответствующих элементов четной и нечетной последовательностей: {summed_elements}")
print(f"Минимальный элемент полученной последовательности: {min_element}")
```

Протокол работы программы:

Последовательность A: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

Четные элементы последовательности A: [2, 4, 6, 8, 10]

Нечетные элементы последовательности A: [1, 3, 5, 7, 9]

Сумма соответствующих элементов четной и нечетной последовательностей: [3, 7, 11, 15, 19]

Минимальный элемент полученной последовательности: 3

Process finished with exit code 0

Постановка задачи №2.

Из списка: ['Валентин', 'Петр', 'Анна', 'Евгений', 'Константин', 'Валерия', 'Юлия'] получить новый список, в котором длина слов не превышает 5 символов

Текст программы:

```
def get_word_length(word):  
    return len(word)  
  
# фильтрации слов по длине  
def filter_words_by_length(words, max_length):  
    return list(filter(lambda word: get_word_length(word) <= max_length, words))  
  
names = ['Валентин', 'Петр', 'Анна', 'Евгений', 'Константин', 'Валерия', 'Юлия']  
  
# фильтруем список имен по длине слов, не превышающей 5 символов  
filtered_names = filter_words_by_length(names, 5)  
  
print(' '.join(filtered_names))
```

Протокол работы программы:

Петр Анна Юлия

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием списковых включений, итераторов, генераторов в IDE PyCharm Community.