

Практическое занятие № 13

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: Из последовательности на n целых чисел создать новую последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов.

Текст программы:

```
# Из последовательности на n целых чисел создать новую последовательность, в
# которой каждый последующий элемент равен квадрату суммы двух соседних элементов.
# Вариант 5

q = list(map(int, input('введите последовательность через пробел\n').split())) # вводим последовательность и
# переводим в int каждый элемент
print(q[1] ** 2, * list((q[num - 1] + q[num + 1]) ** 2 for num in range(1, len(q) - 1)), q[-2] ** 2) # вычисляем
# результат и выводим
```

Протокол работы программы:

```
C:\Users\michael.wilds\AppData\Local\Microsoft\Windows\Terminal\powershell.exe
введите последовательность через пробел
12 23 3412 1243 342 123
529 11723776 1602756 14092516 1865956 116964
```

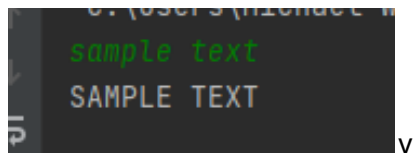
Постановка задачи: Составить генератор (yield), который переведет символы строки из нижнего регистра в верхний.

Текст программы:

```
# Составить генератор (yield), который переведет символы строки из нижнего
# регистра в верхний.
# Вариант 5
def genpz(w): # генератор
    yield w.upper()

print(next(genpz(input())))) # применяется генератор
```

Протокол работы программы:



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