

## Практическое занятие №6

**Тема:** Составление программ со списками 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 задачи №1:** Дан первый член  $A$  и разность  $D$  арифметической прогрессии. Сформировать и вывести список размера 10, содержащий 10 первых членов данной прогрессии:  $A$ ,  $A + D$ ,  $A + 2 \cdot D$ ,  $A + 3 \cdot D$ , ... .

**Тип алгоритма:** циклический.

**Текст программы:**

```
def arithmetic_progression():
    try:
        A = int(input("Введите первый член прогрессии: "))
        D = int(input("Введите разность прогрессии: "))
        lst = []
        for i in range(10):
            lst.append(A + i*D)
        print("Первые 10 членов прогрессии: ", lst)
    except ValueError:
        print("Ошибка: введено некорректное значение")

arithmetic_progression()
```

**Протокол работы программы:**

Введите первый член прогрессии: 1

Введите разность прогрессии: 9

Первые 10 членов прогрессии: [1, 10, 19, 28, 37, 46, 55, 64, 73, 82]

Process finished with exit code 0

**Постановка задачи №2:** Дан список размера N. Найти номер его первого локального минимума (локальный минимум — это элемент, который меньше любого из своих соседей).

**Тип алгоритма:** циклический.

**Текст программы:**

```
def find_local_min():
    try:
        N = int(input("Введите размер списка: "))
        if N == 0:
            print("Ваш список пуст")
            return

        lst = []
        for i in range(N):
            lst.append(int(input("Введите элемент списка: ")))

        # Проверка первого элемента
        if N == 1 or lst[0] < lst[1]:
            print("Номер первого локального минимума: 0")
            return

        # Проверка последнего элемента
        if lst[N - 1] < lst[N - 2]:
            print(f"Номер первого локального минимума: {N - 1}")
            return

        # Проверка элементов в середине списка
        for i in range(1, N - 1):
```

```

        if lst[i] < lst[i - 1] and lst[i] < lst[i +
1]:
            print(f"Номер первого локального минимума:
{i}")
            return

    print("Локальный минимум не найден")

except ValueError:
    print("Ошибка: введено некорректное значение")

find_local_min()

```

### Протокол работы программы:

Введите размер списка: 7

Введите элемент списка: 8

Введите элемент списка: 1

Введите элемент списка: 9

Введите элемент списка: 7

Введите элемент списка: 6

Введите элемент списка: 3

Введите элемент списка: 4

Номер первого локального минимума: 1

Process finished with exit code 0

**Постановка задачи №3:** Дан список размера N (N — четное число). Поменять местами его первый элемент со вторым, третий — с четвертым и т. д.

**Тип алгоритма:** циклический.

### Текст программы:

```
def swap_pairs():
    try:
        N = int(input("Введите размер списка (четное
число): "))
        if N % 2:
            raise ValueError("Размер списка должен быть
четным")
        if N < 0:
            raise ValueError("Размер списка не может быть
отрицательным числом")
        lst = []
        for i in range(N):
            lst.append(int(input("Введите элемент списка:
")))
        for i in range(0, N-1, 2):
            lst[i], lst[i+1] = lst[i+1], lst[i]
        print("Измененный список:", lst)
    except ValueError as error:
        print("Ошибка:", error)

swap_pairs()
```

### Протокол работы программы:

Введите размер списка (четное число): 6

Введите элемент списка: 1

Введите элемент списка: 5

Введите элемент списка: 4

Введите элемент списка: 3

Введите элемент списка: 7

Введите элемент списка: 9

Измененный список: [5, 1, 3, 4, 9, 7]

Process finished with exit code 0

### **Вывод:**

В процессе выполнения практического занятия №6, я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ со списками в IDE PyCharm Community.