

## 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:  
 $A, A + D, A + 2 * D, A + 3 * D, \dots$
2. Дан список размера  $N$ . Найти номер его первого локального минимума (локальный минимум — это элемент, который меньше любого из своих соседей). Дан список размера  $N$  ( $N$  — четное число). Поменять местами его первый элемент со вторым, третий — с четвертым и т. д.
- 3.

**Тип алгоритма:** линейная

Текст программы:

1)

```
1 usage
def progressions(a: int, b: int) -> list:
    progression = []

    for i in range(10):
        member = A + i * D
        progression.append(member)

    return progression

if __name__ == "__main__":
    A = int(input("Введите первый член: "))
    D = int(input("Введите разность арифметической прогрессии: "))

    print(f"Прогрессия первых 10 членов: {progressions(a=A, b=D)}")
```

Введите первый член: 2

Введите разность арифметической прогрессии: 5

Прогрессия первых 10 членов: [2, 7, 12, 17, 22, 27, 32, 37, 42, 47]

Process finished with exit code 0

2)

```
usage
def find_local_minimum(numbers: list) -> int:
    n = len(numbers)

    for i in range(n):
        if i == 0: # Проверяем первый элемент
            if numbers[i] < numbers[i + 1]:
                return i
        elif i == n - 1: # Проверяем последний элемент
            if numbers[i] < numbers[i - 1]:
                return i
        else: # Проверяем остальные элементы
            if numbers[i] < numbers[i - 1] and numbers[i] < numbers[i + 1]:
                return i

    return -1 # Если локальный минимум не найден

if __name__ == "__main__":
    numbers = list()
    try:
        numbers = [int(i) for i in input("Введите числа через пробел: ").split()]

        index = find_local_minimum(numbers)
        if index != -1:
            print("Номер первого локального минимума:", index)
        else:
            print("Локальный минимум не найден")
    except ValueError:
        print("Требуется ввести числа!")
```

Введите числа через пробел: 1 2 3 4 5 6 7 8 9 10

Номер первого локального минимума: 0

Process finished with exit code 0

3)

```

usage
def swap_elements(input_list: list):
    for i in range(0, len(input_list), 2):
        if i+1 < len(input_list):
            input_list[i], input_list[i+1] = input_list[i+1], input_list[i]

if __name__ == "__main__":
    numbers = list()
    try:
        numbers = [int(i) for i in input("Введите числа через пробел: ").split()]
        if len(numbers) % 2 != 0:
            raise KeyError

        swap_elements(numbers)
        print(numbers)
    except ValueError:
        print("Требуется вводить числа!")
    except KeyError:
        print("Количество чисел должно быть четным!")

```

Введите числа через пробел: 1 2 3 4 5 6 7 8 9 10

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

Process finished with exit code 0

### Вывод:

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