

**Тема:** составление программ с использованием списков в IDE PyCharm Community.

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием списков в IDE PyCharm Community.

**Постановка задачи.**

1. Даны целые числа A и B. Сформировать и вывести целочисленный список размера 10, первый элемент которого равен A, второй равен B, а каждый последующий элемент равен сумме всех предыдущих.
2. Дан список размера N. Найти максимальный из его локальных минимумов (локальный минимум - это элемент, который меньше любого из своих соседей).
3. Дан список размера N. Возвести в квадрат все его локальные минимумы (то есть числа, меньшие своих соседей).

**Тип алгоритма:** ветвящийся, линейная.

Текст программы 1:

```
#B6 Даны целые числа A и B. Сформировать и вывести целочисленный список
#размера 10, первый элемент которого равен A, второй равен B, а каждый последующий
#элемент равен сумме всех предыдущих.

def generate_list(a, b):
    result = [a, b]
    for i in range(1, 10):
        next_elem = a + b
        result.append(next_elem)
        a, b = b, next_elem
    return result

A = int(input("Введите число A: "))
B = int(input("Введите число B: "))

if A > 2 and B > 2:
    lst = generate_list(A, B)
    print("Сформированный список: ", lst)
else:
    print("Числа A и B должны быть больше 2.")
```

Текст программы 2:

```
def find_local_minima(lst):
    n = len(lst)
    if n < 3:
        return []

    minima_list = []

    for i in range(1, n - 1):
        if lst[i - 1] > lst[i] < lst[i + 1]:
            minima_list.append(lst[i])

    return max(minima_list)

original_list = [4, 2, 1, 3, 5, 1, 7, 2, 8]
local_minima_list = find_local_minima(original_list)

print("Исходный список:", original_list)
print("Максимальный локальный минимум:", local_minima_list)
```

Текст программы 3:

```
def find_local_minima(lst):
    n = len(lst)
    if n < 3:
        return []

    minima_list = []

    for i in range(1, n - 1):
        if lst[i - 1] > lst[i] < lst[i + 1]:
            minima_list.append(lst[i] ** 2)

    return minima_list

original_list = [4, 2, 1, 3, 5, 1, 7, 2, 8]
local_minima_list = find_local_minima(original_list)

print("Исходный список:", original_list)
print("Локальные минимумы:", local_minima_list)
```

**Вывод:** закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрёл навыки составления программ ветвящийся и линейной структуры с использованием списков в IDE PyCharm Community.