

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:

```
#B6 Дан список размера N. Найти максимальный из его локальных
#минимумов (локальный минимум - это элемент, который меньше любого из своих соседей).

def find_local_min(lst):
    local_mins = []
    for i in range(1, len(lst) - 1):
        if lst[i] < lst[i-1] and lst[i] < lst[i+1]:
            local_mins.append(lst[i])
    return max(local_mins)

lst = [3, 2, 5, 1, 4, 6]
max_local_min = find_local_min(lst)
print(max_local_min)
```

Текст программы 3:

```
#B6 Дан список размера N. Возвести в квадрат все его локальные минимумы
#(то есть числа, меньшие своих соседей).

def square_local_min(lst):
    result = []
    for i in range(1, len(lst) - 1):
        if lst[i] < lst[i-1] and lst[i] < lst[i+1]:
            result.append(lst[i]**2)
    return result

lst = [3, 2, 5, 1, 4, 6]
squared_local_mins = square_local_min(lst)
print(squared_local_mins)
```

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.