

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: $A, A + D, A + 2*D, A + 3*D, \dots$.

Текст программы 1

```
A = input("Введите первый член арифметической прогрессии: ")
D = input("Введите разность арифметической прогрессии: ")

while type(A) != float:
    try:
        A = float(A)
    except ValueError:
        print("Некорректно введены данные!")
        A = input("Введите первый член арифметической прогрессии: ")

while type(D) != float:
    try:
        D = float(D)
    except ValueError:
        print("Некорректно введены данные!")
        D = input("Введите разность арифметической прогрессии: ")

list_arif = []
for i in range(0, 10):
    list_arif.append(A + i * D)

print(list_arif)
```

Протокол работы программы 1

Введите первый член арифметической прогрессии: 5

Введите разность арифметической прогрессии: 4

[5.0, 9.0, 13.0, 17.0, 21.0, 25.0, 29.0, 33.0, 37.0, 41.0, 45.0]

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).

Текст программы 2

```
def find_first_local_minimum(lst):
    if not lst:
        return None

    for i in range(len(lst)):
        if i == 0 and len(lst) > 1 and lst[i] < lst[i + 1]:
            return i
        elif i == len(lst) - 1 and len(lst) > 1 and lst[i] < lst[i - 1]:
            return i
        elif 0 < i < len(lst) - 1 and lst[i] < lst[i - 1] and lst[i] < lst[i
+ 1]:
            return i

    return None

def main():
    try:
        N = int(input("Введите размер списка N: "))
        if N <= 0:
            print("Размер списка должен быть положительным числом.")
            return

        print(f"Введите {N} элементов списка через пробел:")
        lst = list(map(int, input().split()))

        if len(lst) != N:
            print(f"Ожидалось {N} элементов, но введено {len(lst)}.")
            return

        result = find_first_local_minimum(lst)
        if result is not None:
            print(f"Номер первого локального минимума: {result}")
        else:
            print("Локальный минимум не найден.")

    except ValueError:
        print("Ошибка: введите корректные числа.")
    except Exception as e:
        print(f"Произошла ошибка: {e}")

main()
```

Протокол работы программы 2

Введите размер списка N: 5

Введите 5 элементов списка через пробел:

1 2 3 4 5

Номер первого локального минимума: 0

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.

Текст программы 3

```
def swap_pairs(lst):
    if len(lst) % 2 != 0:
        raise ValueError("Список должен иметь четное количество элементов.")

    for i in range(0, len(lst), 2):
        lst[i], lst[i + 1] = lst[i + 1], lst[i]

    return lst

def main():
    try:
        n = int(input("Введите четное число N (размер списка): "))
        if n <= 0:
            print("Размер списка должен быть положительным числом.")
            return
        if n % 2 != 0:
            print("N должно быть четным числом.")
            return

        print(f"Введите {n} элементов списка через пробел:")
        lst = list(map(int, input().split()))

        if len(lst) != n:
            print(f"Ожидалось {n} элементов, но введено {len(lst)}.")
            return

        result = swap_pairs(lst)
        print("Список после обмена элементов попарно:", result)

    except ValueError as e:
        print(f"Ошибка: {e}")
    except Exception as e:
        print(f"Произошла ошибка: {e}")

main()
```

Протокол работы программы 3

Введите четное число N (размер списка): 4

Введите 4 элементов списка через пробел:

2 3 4 5

Список после обмена элементов попарно: [3, 2, 5, 4]

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработала навыки составления списков в IDE PyCharm Community. Выполнены разработка кода,

отладка, тестирование, оптимизация программного кода. Готовый программный код выложен на GitHub.