

## Практическое занятие №6

**Тема:** Составление программ со списками в IDE PyCharm Community.

### Цели практического занятия:

Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со списками в IDE PyCharm Community.

### Постановка задачи 1:

Дан целочисленный список размера 10. Вывести все содержащиеся в данном списке нечетные числа в порядке возрастания их индексов, а также их количество K.

### Текст программы № 1:

```
# Дан целочисленный список размера 10.
# Вывести все содержащиеся в данном списке нечетные числа
в порядке возрастания их индексов,
# а также их количество K.
import random # Импортируем функцию random
mas = [] # Создаём список, в котором будут начальные
данные
s = [] # Создаём список, в котором будут конечные
результаты
k = 0 # Счётчик суммы нечётных чисел
for i in range(10): # Цикл проходит n количество раз
    mas.append(random.randint(1, 50)) # Добавляется в
список mas n количество чисел
print("В список занеслось 10 случайных чисел: ")
print(*mas)
for i, x in enumerate(mas): # Проходит по массиву mas,
enumerate возвращает индекс и число в списке
    if x % 2 != 0: # Проверяем на нечётность
        print("Элемент №", i, "это: ", x)
        k += 1 # Счётчик суммы нечётных чисел
        s.append(x) # Заносим нечётные числа в новый
массив
print("Числа в порядке возрастания их индексов: ")
```

```
print(*s) # Выводим конечный список в порядке
возрастания их индексов
print(f"Количество нечетных чисел: {k}") # Вывод
количества нечётных чисел
```

### Протокол работы программы №1:

```
В список занеслось 10 случайных чисел:
46 10 17 5 8 46 25 19 12 32
Элемент № 2 это: 17
Элемент № 3 это: 5
Элемент № 6 это: 25
Элемент № 7 это: 19
Числа в порядке возрастания их индексов:
17 5 25 19
Количество нечетных чисел: 4

Process finished with exit code 0
```

### Постановка задачи 2:

Дан список размера N. Найти минимальный из его локальных максимумов (локальный максимум - число, которое будет больше обоих из своих соседей).

### Текст программы №2:

```
# Дан список размера N.
# Найти минимальный из его локальных максимумов.
# Локальный максимум - число, которое будет больше обоих
из своих соседей.
import random # Импортируем функцию рандом.
mas = [] # Создаём список, в котором будут начальные
данные
s = [] # Создаём список, в котором будут конечные
результаты
n = input("Введите количество чисел в списке: ")
while True: # Программа работает, даже при ошибке
    try: # Пользователь ввёл число
        n = int(n)
```

```
        break # Досрочно прерывает цикл
    except ValueError: # Проверка исключений (пользователь
ввел не число)
        print("Введите число, пожалуйста!")
    n = input("Введите количество чисел в списке: ")
    for j in range(n): # Цикл проходит n количество раз
        mas.append(random.randint(0, 50)) # Добавляем в
список mas n количество чисел
    print(f"В список занеслось {n} случайных чисел: ")
    print(*mas)
    mas = [-1] + mas + [-1] # Расширяем список, чтобы не
возникало ошибок при проверке крайних чисел
    for i in range(1, len(mas) - 1): # Проходим по списку от
2 элемента до предпоследнего (не считает крайние -1)
        if mas[i] > mas[i - 1]: # Проверяем на локальный
минимум
            if mas[i] > mas[i + 1]: # Проверяем на локальный
минимум
                s.append(mas[i]) # Заносим в список s
локальные минимумы

    print("В списке найдены следующие локальные максимумы: ")
    print(*s) # Вывод самого списка локальных максимумов
    print("Наименьший из локальных максимумов = ", min(s)) #
В списке s с помощью функции min находим минимальный
элемент.
```

## Протокол работы программы №2:

```
Введите количество чисел в списке: love
Введите число, пожалуйста!
Введите количество чисел в списке: 11
В список занеслось 11 случайных чисел:
7 40 4 33 23 38 47 47 10 8 19
В списке найдены следующие локальные максимумы:
40 33 19
Наименьший из локальных максимумов = 19

Process finished with exit code 0
```

### Постановка задачи 3:

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

### Текст программы №3:

```
# Дан список размера N.
# Возвести в квадрат все его локальные минимумы (то есть
числа, меньшие своих соседей) .
import random # Импортируем функцию random
mas = [] # Создаём список, куда будут внесены случайные
числа
s = [] # Создаём вспомогательный списка
mas1 = [] # Создаём список для конечного результата
n = input("Введите количество чисел в списке: ")
while True: # Программа работает, даже при ошибке
    try: # Пользователь ввёл число
        n = int(n)
        break # Досрочно прерывает цикл
    except ValueError: # Проверка исключений (пользователь
ввел не число)
        print("Введите число, пожалуйста!")
        n = input("Введите количество чисел в списке: ")
for j in range(n): # Цикл проходит n количество раз
    mas.append(random.randint(0, 20)) # Добавляется в
список mas n количество чисел
print(f"В список занеслось {n} случайных чисел: ")
print(*mas)
mas = [500] + mas + [500] # Расширяем список, чтобы не
возникало ошибок при проверке крайних чисел
for i in range(1, len(mas) - 1): # Проходим по списку от
2 элемента до предпоследнего (не считает крайние 500)
    if mas[i - 1] > mas[i] and mas[i + 1] > mas[i]: #
Проверка на локальный минимум
        s.append(mas[i]) # Заносим в список s локальные
минимумы
        s = [500] + s + [500] # Расширяем список, чтобы
не возникало ошибок при проверке крайних чисел
        for g in range(1, len(s) - 1): # Проходим по
списку от 2 элемента до предпоследнего (не считает
крайние 500)
```

```
        if s[g - 1] > s[g] and s[g + 1] > s[g]: #
Сравнение элементов массивов
        if mas[i] == s[g]: # Если элементы равны,
то..
        mas1.append(mas[i] * mas[i]) # ..
возводим в квадрат элемент и добавляем в конечный список
    else: # Если число не локальный минимум то..
        mas1.append(mas[i]) # ..заносим его в список

print("Список с локальными минимумами возведенных в
квадрат: ")
print(*mas1) # Вывод конечного списка.
```

### Протокол работы программы №3:

```
Введите количество чисел в списке: love
Введите число, пожалуйста!
Введите количество чисел в списке: 10
В список занеслось 10 случайных чисел:
18 13 5 10 15 11 0 19 9 18
Список с локальными минимумами возведенных в квадрат:
18 13 25 10 15 11 0 19 81 18

Process finished with exit code 0
```

### Вывод:

В ходе данной практической работы закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ со списками в IDE PyCharm Community.

Были использованы языковые конструкции for, if, else, s = [], mas = [].

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.