

Практическое занятие №6

Тема: составление программ со списками в IDE PyCharm Community.

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, составление программ со списками в IDE PyCharm Community.

Постановка задачи.

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:

```
# 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
a = [] # создаем список, в который будут конечные результаты
k = 0 # счётчик суммы нечетных чисел
for i in range(10): # цикл проходит n количество раз
    mas.append(random.randint(0, 100)) # 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 само число в списке
    if x % 2 != 0: # проверка на нечётность
        print('Элемент №', i, 'это:', x)
        k += 1 # счетчик суммы нечетных чисел
        a.append(x) # заносим нечетные числа в новый массив
print('Числа в порядке возрастания их индексов:')
print(*a) # выводим конечный список в порядке возрастания их индексов.
print(f'Количество нечетных чисел равно = {k}') # вывод количества нечётных
чисел
```

Протокол работы программы:

В список занеслось 10 случайных чисел:

7 26 71 55 5 55 37 82 50 62

Элемент № 0 это: 7

Элемент № 2 это: 71

Элемент № 3 это: 55

Элемент № 4 это: 5

Элемент № 5 это: 55

Элемент № 6 это: 37

Числа в порядке возрастания их индексов:

7 71 55 5 55 37

Количество нечетных чисел равно = 6

Process finished with exit code 0

Постановка задачи.

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

Текст программы:

```
# Дан список размера N. Найти минимальный из его локальных максимумов.
# Локальный максимум - число, которое будет больше обоих из своих соседей.
import random # импортируем функцию random.
mas = [] # создаём список, в котором будут начальные данные
a = [] # создаем список, в который будут конечные результаты
n = int(input('Введите количество чисел в списке: '))
for j in range(n): # цикл проходит n количество раз
    mas.append(random.randint(0, 100)) # добавляется 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
проверки 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]: # проверка на локальный минимум
        if mas[i] > mas[i + 1]: # проверка на локальный минимум
            a.append(mas[i]) # заносим в список a локальные минимумы

print('В списке найдены следующие локальные максимумы: ')
print(*a) # вывод самого списка локальных максимумов
print('Наименьший из локальных максимумов = ', min(a)) # в списке a с помощью
функции мин ищем минимальный элемент.
```

Протокол работы программы:

Введите количество чисел в списке: 10

В список занеслось 10 случайных чисел:

51 79 40 20 84 99 18 2 57 33

В списке найдены следующие локальные максимумы:

79 99 57

Наименьший из локальных максимумов = 57

Process finished with exit code 0

Постановка задачи.

Дан список размера N. Возвести в квадрат все его локальные минимумы.
Локальный минимум - число, которое будет меньше обоих из своих соседей.

Текст программы:

```
# Дан список размера N. Возвести в квадрат все его локальные минимумы
# Локальный минимум - число, которое будет меньше обоих из своих соседей.
import random # импортируем функцию random.
mas = [] # создание списка, куда будут внесены случайные числа
a = [] # создание вспомогательного списка
mas1 = [] # создание списка для конечного результата
n = int(input('Введите количество чисел в списке: '))
for j in range(n): # цикл проходит n количество раз
    mas.append(random.randint(1, 100)) # добавляется в список mas n количество
чисел
print(f'В список занеслось {n} случайных чисел: ')
print(*mas)
mas = [1000] + mas + [1000] # расширяем список, чтобы не возникало ошибок при
проверки крайних чисел
for i in range(1, len(mas) - 1): # проходим по списку от 2 элемента до
предпоследнего (не считает крайние 1000)
    if mas[i - 1] > mas[i] and mas[i + 1] > mas[i]: # проверка на локальный
минимум
        a.append(mas[i]) # заносим в список a локальные минимумы
        a = [1000] + a + [1000] # расширяем список, чтобы не возникало ошибок
при проверки крайних чисел
        for p in range(1, len(a) - 1): # проходим по списку от 2 элемента до
предпоследнего (не считает крайние 1000)
            if a[p - 1] > a[p] and a[p + 1] > a[p]: # сравнение элементов
массивов
                if mas[i] == a[p]: # 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, возведенными в квадрат')
print(*mas1) # вывод конечного списка.
```

Протокол работы программы:

Введите количество чисел в списке: 15

В список занеслось 15 случайных чисел:

91 78 53 85 9 11 94 61 12 85 31 20 2 34 99

Список с локальными минимумами, возведенными в квадрат

91 78 2809 85 81 11 94 61 144 85 31 20 4 34 99

Process finished with exit code 0

Вывод: В процессе выполнения шестого практического занятия я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, составление программ со списками в IDE PyCharm Community.

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

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

Готовые программные коды выложены на GitHub.