

Практическое занятие № 6

Тема: составление программ с функциями 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: написать программу, выводящую все содержащиеся в 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.
'''
numbers = []

print("Введите 10 целых чисел:")
for i in range(10):
    num = input(f"Введите число {i + 1}: ")
    while type(num) != int:
        try:
            num = int(num)
        except ValueError:
            print("Некорректный ввод. Пожалуйста, введите целое число.")
            num = input(f"Введите число {i + 1} еще раз: ")
    numbers.append(num)

odd_numbers = []

for num in numbers:
    if num % 2 != 0:
        odd_numbers.append(num)

print("Нечетные числа в порядке возрастания их индексов:")
for num in odd_numbers:
    print(num, end=" ")

K = len(odd_numbers)
print(f"\nКоличество нечетных чисел: {K}")
```

Протокол программы:

Введите 10 целых чисел:

Введите число 1:

1

Введите число 2:

2

Введите число 3:

3

Введите число 4:

4

Введите число 5:

5

Введите число 6:

6

Введите число 7:

7

Введите число 8:

8

Введите число 9:

9

Введите число 10:

0

Нечетные числа в порядке возрастания их индексов:

1 3 5 7 9

Количество нечетных чисел: 5

Постановка задачи: написать программу, находящую минимальный из его локальных максимумов в списке (локальный минимум — это элемент, который меньше любого из своих соседей)

Текст программы:

```
'''
Дан список размера N. Найти минимальный из его локальных максимумов
(локальный минимум – это элемент, который меньше любого из своих соседей).
'''
import random

n = input('Введите длину списка: ')
while type(n) != int:
    try:
        n = int(n)
    except ValueError:
        print("Некорректный ввод. Пожалуйста, введите целое число.")
        n = input('Введите длину списка: ')

lst = [random.randint(a=1, b=100) for _ in range(n)]

print("Сгенерированный список:", lst)

local_minima = []

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

if local_minima:
    min_local_minimum = min(local_minima)
    print("Минимальный из локальных минимумов:", min_local_minimum)
else:
    print("Локальных минимумов не найдено.")
```

Протокол программы:

Введите длину списка:

Сгенерированный список: [58, 47, 80, 62, 35, 57, 72, 4, 69, 54, 8, 1, 25, 90, 5, 43, 96, 26, 41, 59]

Минимальный из локальных минимумов: 1

Постановка задачи: написать программу, возводящую в квадрат все локальные минимумы списка (то есть числа, меньшие своих соседей).

Текст программы:

```

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

n = input('Введите длину списка: ')
while type(n) != int:
    try:
        n = int(n)
    except ValueError:
        print("Некорректный ввод. Пожалуйста, введите целое число.")
        n = input('Введите длину списка: ')

lst = [random.randint(a: 1, b: 100) for _ in range(n)]

print("Сгенерированный список:", lst)

usage
def square_local_minima(lst):
    if len(lst) < 3:
        return lst

    result = lst[:]

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

squared_minima = square_local_minima(lst)
print(f'Локальные минимумы в квадрате: {squared_minima}')

```

Протокол программы:

Введите длину списка: 10

Сгенерированный список: [58, 56, 98, 79, 40, 97, 35, 5, 7, 46]

Локальные минимумы в квадрате: [58, 3136, 98, 79, 1600, 97, 35, 25, 7, 46]

Минимальный из локальных минимумов: 1

Вывод: я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с функциями в IDE PyCharm Community. Все программы и отчет выложены на GitHub.