

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

Тип алгоритма: циклический

Текст программы:

```
PZ > PZ-6 > pz6.1.py > ...
1  # Дан целочисленный список размера 10. Вывести все содержащиеся в данном списке
2  # четные числа в порядке убывания их индексов, а также их количество K.
3
4  import random
5
6  lst = [int(random.randint(1, 100)) for i in range(10)]
7
8  even_numbers = []
9  # Проходим по списку с конца
10 for i in lst[::-1]:
11     if i % 2 == 0: # Проверяем, является ли число четным
12         even_numbers.append(i)
13
14 print("Четные числа в порядке убывания индексов:", even_numbers)
15 print("Количество четных чисел:", len(even_numbers))
```

Протокол работы программы:

Четные числа в порядке убывания индексов: [36, 28, 34, 84, 34]

Количество четных чисел: 5

Process finished with exit code 0

2. Дан список размера N. Найти количество участков, на которых его элементы монотонно возрастают.

Тип алгоритма: циклический

Текст программы:

```
PZ > PZ-6 > pz6.2.py > ...
1  # Дан список размера N. Найти количество участков, на которых его элементы
2  # монотонно возрастают.
3
4  import random
5  n = input("Введите длину случайного списка: ")
6  while type(n) != int:
7      try:
8          n = int(n)
9      except:
10         n = input("Введите длину случайного списка: ")
11
12  lst = [int(random.randint(1, 100)) for i in range(n)]
13  print(lst)
14  # Переменная для подсчета количества участков
15  count = 0
16  n = len(lst)
17
18  # Проверка наличия хотя бы одного участка
19  if n > 0:
20      in_increasing_segment = False
21
22      for i in range(1, n):
23          if lst[i] > lst[i - 1]: # Если элемент больше предыдущего
24              if not in_increasing_segment:
25                  count += 1 # Начинаем новый участок
26                  in_increasing_segment = True
27              else:
28                  in_increasing_segment = False
29
30  print("Количество участков монотонно возрастающих чисел:", count)
```

Протокол работы программы:

Введите длину случайного списка: 10

[14, 7, 77, 61, 71, 71, 95, 26, 62, 17]

Количество участков монотонно возрастающих чисел: 4

Process finished with exit code 0

3. Дан список размера N. Заменить каждый элемент списка на среднее арифметическое этого элемента и его соседей.

Тип алгоритма: циклический

Текст программы:

```

PZ > PZ-6 > pz6.3.py > ...
1  # Дан список размера N. Заменить каждый элемент списка на среднее арифметическое
2  # этого элемента и его соседей.
3
4  import random
5  n = input("Введите длину случайного списка: ")
6  while type(n) != int:
7      try:
8          n = int(n)
9      except:
10         n = input("Введите длину случайного списка: ")
11
12  lst = [int(random.randint(1, 100)) for i in range(n)]
13
14  new_lst = []
15
16  # Для первого элемента
17  new_lst.append((lst[0] + lst[1]) / 2)
18
19  # Для элементов между первым и последним
20  for i in range(1, len(lst) - 1):
21      new_lst.append((lst[i - 1] + lst[i] + lst[i + 1]) / 3)
22
23  # Для последнего элемента
24  new_lst.append((lst[-2] + lst[-1]) / 2)
25
26  print("Новый список:", new_lst)

```

Протокол работы программы:

Введите длину случайного списка: 10

Новый список: [27.5, 36.666666666666664, 37.333333333333336, 41.333333333333336, 47.666666666666664, 65.33333333333333, 60.666666666666664, 48.333333333333336, 43.333333333333336, 41.0]

Process finished with exit code 0

Вывод: выработали первичные навыки работы с IDE PyCharm Community, навыки составления программ со списками