

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: создать программу, которая выводит все элементы списка с четными номерами в порядке возрастания номеров без использования условного оператора

Тип алгоритма: последовательный

Текст программы:

```
# Составить функцию, которая напечатает 40 любых символов
# Дан список A размера N (N — четное число). Вывести его элементы с четными
# номерами в порядке возрастания номеров: A2, A4, A6, ..., AN. Условный оператор не
# использовать.
```

```
element = 1
```

```
N = int(input('Введите четное число - размер списка A: '))
```

```
while True:
```

```
    if N % 2 != 0:
```

```
        print('Число нечетное!')
```

```
        N = int(input('Введите четное число: '))
```

```
    break
```

```
A = list(range(N))
```

```
print(A[::2])
```

Протокол работы программы:

Введите четное число - размер списка A: 10

[0, 2, 4, 6, 8]

Process finished with exit code 0

Задание 2

Постановка задачи: найти элемент списка, который наиболее близок к числу R

Тип алгоритма: последовательный

Текст программы:

```
# Дано число R и список A размера N. Найти элемент списка, который наиболее близок  
# к числу R (то есть такой элемент AK, для которого величина |AK - R| является  
# минимальной).
```

```
import random
```

```
N = int(input('Введите размер списка A: '))
```

```
R = int(input('Введите число: '))
```

```
A = []
```

```
num = random.randint(0, R * 10)
```

```
while len(A) != N:
```

```
    A.append(num)
```

```
    num = random.randint(0, R * 10)
```

```
print(f'список A - {A}')
```

```
element = A[0]
```

```
for i in A:
```

```
    min_num = abs(i - R)
```

```
    max_num = i - R
```

```
    if max_num < min_num:
```

```
        min_num = max_num
```

```
        element = i
```

```
print(f'Наиболее близкий элемент списка к числу R: {element}')
```

Протокол работы программы:

Введите размер списка A: 10

Введите число: 11

список A - [44, 52, 62, 65, 50, 58, 0, 87, 29, 18]

Наиболее близкий элемент списка к числу R: 0

Process finished with exit code 0

Задание 3

Постановка задачи: Осуществить сдвиг элементов списка влево на одну позицию (при этом A_N перейдет в A_{N-1} , A_{N-1} — в A_{N-2} , ..., A_2 — в A_1 , а исходное значение первого элемента будет потеряно)

Тип алгоритма: циклический

Текст программы:

```
# Дан список размера N. Осуществить сдвиг элементов списка влево на одну позицию
# (при этом  $A_N$  перейдет в  $A_{N-1}$ ,  $A_{N-1}$  — в  $A_{N-2}$ , ...,  $A_2$  — в  $A_1$ , а исходное значение
# первого элемента будет потеряно). Последний элемент полученного списка
# положить равным 0.
```

```
import random
```

```
N = int(input('Введите размер списка: '))
spisok = []
while len(spisok) != N:
    spisok.append(random.randint(0, N * 10))
print(f'Изначальный список: {spisok}')
spisok.pop(0)
spisok[-1] //= spisok[-1]
for i in spisok:
    i -= 1
    print(i)
```

Вывод: в процессе выполнения практического занятия выработал навыки составления программ со списками в IDE PyCharm Community. Были использованы языковые конструкции while, if, else, for. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.