

Отчёт

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.

Разработать программу, решающую задачу:

Дан список A размера N. Вывести вначале его элементы с чётными номерами (в порядке возрастания номеров), а затем - элементы с нечётными номерами (также в порядке возрастания номеров): A₂, A₄, A₆, ..., A₁, A₃, A₅, Условный оператор не использовать

Тип алгоритма 1: циклический

Текст программы 1:

```
#Вариант 13. Дан список A размера N. Вывести вначале его элементы с чётными номерами
```

```
 #(в порядке возрастания номеров), а затем - элементы с нечётными номерами (также в
```

```
 #порядке возрастания номеров): A2, A4, A6, ..., A1, A3, A5, .... Условный оператор не использовать
```

```
from random import randint
```

```
try:
```

```
    n = int(input("Введите размер списка "))
```

```
except Exception:
```

```
    print("Вы ввели что-то не то")
```

```
mylist = []
```

```
for i in range(1, n+1):
```

```
mylist.append(randint(1,100))  
print(f"Исходный список {mylist}")  
print("Элементы с нечётными номерами: ")  
for i in range(0, n, 2):  
    print(mylist[i])  
print("Элементы с чётными номерами: ")  
for i in range(1, n, 2):  
    print(mylist[i])
```

Протокол работы программы 1:

Введите размер списка 7

Исходный список [62, 51, 60, 99, 26, 23, 22]

Элементы с нечётными номерами:

62

60

26

22

Элементы с чётными номерами:

51

99

23

Process finished with exit code 0

Постановка задачи 2.

Разработать программу, которая решает задачу:

Дано число R и список размера N. Найти два соседних элемента списка, сумма которых наиболее близка к числу R, и вывести эти элементы в порядке

возрастания их индексов (определение наиболее близких чисел - то есть такой элемент A_k , для которого величина $|A_k - R|$ является минимальной)

Тип алгоритма 2: циклический

Текст программы 2:

#Дано число R и список размера N . Найти два соседних элемента списка, сумма

#которых наиболее близка к числу R , и вывести эти элементы в порядке

#возрастания их индексов (определение наиболее близких чисел - то есть

#такой элемент A_k , для которого величина $|A_k - R|$ является минимальной)

```
from random import randint
```

```
try:
```

```
    n = int(input("Введите размер списка "))
```

```
    r = int(input("Введите число R "))
```

```
except Exception:
```

```
    print("Вы ввели что-то не то")
```

```
list1 = []
```

```
for i in range(1,n+1):
```

```
    list1.append(randint(-100,100))
```

```
print(f"Исходный список {list1}")
```

```
element1, element2 = list1[0], list1[1]
```

```
summa = element1 + element2
```

```
for i in range(1, n - 1):
```

```
    result=abs(list1[i] + list1[i + 1]-r)
```

```
if result < summa:
```

```
    summa=result
```

```
    element1, element2 = list1[i], list1[i + 1]
```

```
print(f"Первый элемент {element1}, второй элемент {element2}")
```

Протокол работы программы 2:

Введите размер списка 6

Введите число R 5

Исходный список [24, 80, -99, -15, -38, -27]

Первый элемент 80, второй элемент -99

Process finished with exit code 0

Постановка задачи 3.

Разработать программу, которая решает задачу:

Дан список размера N и целое число K ($1 < K < N$). Осуществить сдвиг элементов списка влево на K позиций (при этом A_n перейдёт в A_{n-k} , A_{n-1} в A_{n-k-1} , A_{k+1} в A_1 , а исходное значение K первых элементов будет потеряно) последние K элементов полученного списка положить равными 0.

Тип алгоритма 3: циклический

Текст программы 3:

```
#Вариант 13. Дан список размера N и целое число K (1<K<N). Осуществить
```

```
#сдвиг элементов списка влево на K позиций (при этом An перейдёт в An-k,
```

```
#An-1 в An-k-1, Ak+1 в A1, а исходное значение K первых элементов будет потеряно)
```

```
#последние K элементов полученного списка положить равными 0.
```

```
import random
```

```
try:
```

```

N = int(input("Введите размер списка N "))
K = int(input("Введите число позиций сдвига K "))

except Exception:

    print("Вы ввели что-то не то")

list2 = []

for i in range(1,N+1):

    list2.append(random.randint(-100,100))

print(f"Исходный список {list2}")

for i in range(0,N-K):

    list2[i] = list2[i+K]

for i in range(N-K,N):

    list2[i] = 0

print(f"Сдвиг списка {list2}")

```

Протокол работы программы 3:

Введите размер списка N 7

Введите число позиций сдвига K 3

Исходный список [75, 60, -52, 10, 10, -74, 7]

Сдвиг списка [10, 10, -74, 7, 0, 0, 0]

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ со списками в IDE PyCharm

Community. Были использованы цикл `for`, библиотека `random`, оператор `abs`, `range`, была проведена проверка на исключения.

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

Готовые программные коды выложены на [GitHub](#).