

Практическое задание №6.1

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

Тест программы:

```
# Дан список A размера N. Вывести вначале его
элементы с четными номерами (в
# порядке возрастания номеров), а затем — элементы с
нечетными номерами (также в
# порядке возрастания номеров): A2, A4, A6, . . . ,
A1, A3, A5, ... . Условный оператор не
# использовать.
try: # обработчик исключений
    n = int(input('Введите длину списка ')) # ввод
числа
    a = [int(i) for i in input('Введите числа от 1 до
длины вашего списка через пробел ').split()] # цикл
    for i in range(1, n, 2): # цикл перебирает
последовательность чисел
        print(a[i], end=" ") # вывод четных чисел
    print('\n')
    for i in range(0, n, 2): # цикл перебирает
последовательность чисел
        print(a[i], end=" ") # вывод нечетных чисел
except ValueError: # обработчик искл
```

```
print("Ошибка")
```

Протокол работы программы:

```
Введите длину списка 6
Введите числа от 1 до длины вашего списка через пробел 1 2 3 4 5 6
2 4 6

1 3 5
Process finished with exit code 0
```

Практическое задание №6.2

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: 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 AK , для которого величина $|AK - R|$ является минимальной).

Тест программы:

```
# Дано число R и список размера N. Найти два
различных 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 АК, для
# которого величина |АК - R| является минимальной).
try: # обработчик исключений
    from random import randint # добавляем рандом

    N = int(input('Введите размер списка ')) # Ввод
списка
    R = int(input('Введите R ')) # ввод числа R
    x = []
    sam = 0
    index_1 = 0
    index_2 = 0
    v = 0
    y = 100
    for i in range(N): # цикл
        x.append(randint(0, 100))
    print(x)
    x.append(0)
    for i in range(len(x) - 1): # цикл для поиска
чисел
        sam = x[i] + x[i + 1]
        v = abs(R - sam)
        if v <= y: # проверка на истинность
            y = v
            index_1 = x[i]
            index_2 = x[i + 1]
    print('Первое число ', index_1) # вывод первого
числа
    print('Второе число', index_2) # вывод второго
числа
except ValueError:
    print("Ошибка")
```

Протокол работы программы:

```
Введите размер списка 10
Введите R 2
[83, 7, 50, 54, 22, 77, 33, 7, 2, 24]
Первый элемент 7
Второй элемент 2

Process finished with exit code 0
```

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

Практическое задание №6.3

Тема: составление программ со списками в IDE PyCharm Community.

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

Постановка задачи: Дан список размера N и целое число K ($1 < K < N$). Осуществить сдвиг элементов списка влево на K позиций (при этом A_N перейдет 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.

Тест программы:

```
# Дан список размера N и целое число K ( $1 < K < N$ ).  
Осуществить сдвиг элементов  
# списка влево на K позиций (при этом AN перейдет 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.  
try:  
    import random  
  
    import lst as lst  
  
    k = random.randrange(1, 10) # Генерируется  
случайное целое число в заданном промежутке  
    n = random.randrange(k + 1, 34)  
    print("N = ", n)  
  
    a = [i + 1 for i in range(n)] # Генерируется  
список, заполненный числами от 1 до n в порядке  
возрастания  
    print(a)  
    print(a[k:] + a[:k])  
  
    b = []  
    for i in range(0, k): # Формирование списка с  
числами  
        b.append(a[i])  
  
    for i in range(0, n - k): # Числа, которые не  
переносятся, увеличиваются на индекс сдвига  
        a[i] = a[i + k]  
  
    g = 0  
    for i in range(n + k, n): # заносим числа из  
начала  
        a[i] = b[g]  
        g += 1
```

```
print(f"Результат: {lst}") # Вывод списка
except ValueError:
    print("Ошибка")
```

Протокол работы программы:

```
N = 30
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]
[8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 1, 2, 3, 4, 5, 6, 7]
```

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