Студентка группы ИС-25 Ищенко М.В.

Практическое занятие № 6

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 №1

Разработать программу, выполняющую данную задачу: дан список A размера N и целое число K (1 < K < N). Вывести элементы список с порядковыми номерами, кратными K: AK, A2*K, A3*K,... . Условный оператор не использовать.

Тип алгоритма: циклический.

Текст программы:

```
# Дан список A размера N и целое число K (1 < K < N).
# Вывести элементы список с порядковыми номерами, кратными K: AK, A2*K, A3*K,...
.
# Условный оператор не использовать.
A = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150]  # Создаем
список A
N = len(A)                                                              # Заданный размер
списка
K = input('Введи число: ')                                              # Задаём значение
K
while type(K) != int:
    try:
        K = int(K)
    except ValueError:
        print('Неправильно ввели!')
        K = input("Введи K заново: ")
result = []                                 # Создаём пустой список для хранения
результата
for i in range(K, N, K):                # Используем цикл для перебора индексов от K
до N с шагом K
    result.append(A[i])
print(result)                           # Выводим результат
```

Вывод: в процессе выполнения практического занятия выработал(а) навыки составления программ циклической структуры в IDE PyCharm Community. Были использованы языковые конструкции while, try, except, for, append.

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

Протокол работы программы:

Введи число: 2

[30, 50, 70, 90, 110, 130, 150]

Process finished with exit code 0

Постановка задачи №2

Разработать программу, находящую количество промежутков монотонности данного списка (то есть участков, на которых его элементы возрастают или убывают).

Тип алгоритма: циклический.

Текст программы:

```
#Дан список размера N. Найти количество его промежутков монотонности
#(то есть участков, на которых его элементы возрастают или убывают).
def schitaem_ciferki(spisok):
    if len(spisok) < 2:        # Если список пустой или имеет один элемент, воз-
вращаем пустой список
        return 0


    schet = 0           # Счётчик промежутков монотонности
    povishenie = None       # Задаем начальное состояние

    for i in range(1, len(spisok)):
        if spisok[i] > spisok[i - 1]:  # Если текущий элемент больше предыдущего
            if povishenie is False:  # Если до этого был убывающий участок
                schet += 1
            povishenie = True  # Устанавливаем состояние как возрастающее
        elif spisok[i] < spisok[i - 1]:  # Если текущий элемент меньше предыдуще-
го
            if povishenie is True:  # Если до этого был возрастающий участок
                schet += 1
            povishenie = False  # Устанавливаем состояние как убывающее
    if povishenie is not None:      # Если последний участок был монотонным, то
добавляем его
        schet += 1

    return schet
```

```
spisok = [3, 1, 2, 4, 3, 5, 6, 2]  # Примерный список
print('Ваш список: ', spisok)
result = schitaem_ciferki(spisok)
print("Количество промежутков монотонности:", result)
```

Протокол работы программы:

Ваш список: [3, 1, 2, 4, 3, 5, 6, 2]

Количество промежутков монотонности: 5

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал(а) навыки составления программ с функциями в IDE PyCharm Community.

Были использованы языковые конструкции def, if, elif, return.

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

Постановка задачи №3

Разработать программу, осуществляющую сдвиг элементов список вправо на одну позицию (при этом A1 перейдет в A2, A2 — в A3, ..., AN-1 — в AN, а исходное значение последнего элемента будет потеряно). Первый элемент полученного списка положить равным 0.

Тип алгоритма: линейный.

Текст программы:

```
# Дан список размера N. Осуществить сдвиг элементов список вправо на одну позицию
# (при этом A1 перейдет в A2, A2 – в A3, ..., AN-1 – в AN, а исходное значение
последнего элемента будет потеряно).
# Первый элемент полученного списка положить равным 0.
def poshli_napravo(spisok):

    if len(spisok) < 2:        # Если список пустой или имеет один элемент, воз-
вращаем пустой список
        return []

    return [0] + spisok[:-1]  # Создаем новый список, ставим 0 в начало и добав-
ляем все элементы, кроме последнего
```

```
# Пример использования:
A = [1, 2, 3, 4, 5]
print('Ваш список: ', A)
result = poshli_napravo(A)
print("Сдвинули! ", result)
```

Протокол работы программы:

Ваш список: [1, 2, 3, 4, 5]

Сдвинули! [0, 1, 2, 3, 4]

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал(а) навыки составления программ с функциями в IDE PyCharm Community.

Были использованы языковые конструкции def, if, return.

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

Готовые программные коды выложены на GitHub.