

ПЗ_6

Первая часть

Тема: Составления программ со списками

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ со списками.

Постановка задачи.

```
'''Дан список A размера N (N – нечетное число). Вывести его элементы с нечетными номерами в порядке убывания номеров: An, An-2, An-4, ..., A1. Условный оператор не использовать.'''
```

Тип алгоритма: линейный

Текст программы:

Вариант 12

```
# Ввод списка пользователем
A = list(map(int, input("Введите элементы списка через пробел: ").split()))

# Проверка, что размер списка нечетный
assert len(A) % 2 != 0, "Размер списка должен быть нечетным"

# Вывод элементов с нечетными номерами в порядке убывания номеров
print(A[::-2])
```

Вывод: В процессе работы я закрепил усвоенные знания, понятия, научился работать со списками.

Вторая часть

Тема: Составления программ со списками

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ со списками.

Постановка задачи.

```
'''Дан список размера N. Найти два соседних элемента, сумма которых максимальна, и вывести эти элементы в порядке возрастания их индексов.'''
чисел:
(A1, B1, C1) и (A2, B2, C2).'''
```

Тип алгоритма: линейный

Текст программы:

Вариант 12

```
def find_max_sum_elements(lst):
    if len(lst) < 2:
        return None
    max_sum = lst[0] + lst[1]
    max_index = 0
    for i in range(2, len(lst)):
        if lst[i-1] + lst[i] > max_sum:
            max_sum = lst[i-1] + lst[i]
            max_index = i - 1
    return lst[max_index], lst[max_index + 1]

try:
    lst = list(map(int, input("Введите числа, разделенные пробелами: ").split()))
    print(find_max_sum_elements(lst))
except:
    print('Введите число!')
```

Вывод: В процессе работы я закрепил усвоенные знания, понятия, научился работать со списками.

Третья часть

Тема: Составления программ со списками

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ со списками.

Постановка задачи.

```
'''Дан список размера N и целое число K (1 < K < N). Осуществить сдвиг элементов списка вправо на K позиций (при этом A1 перейдет в Ak+1, A2 – в Ak+2, ..AN-k – в AN, а исходное значение K последних элементов будет потеряно). Первые K элементов полученного списка положить равными 0.'''
```

Тип алгоритма: циклический

Текст программы:

Вариант 12

```
import random

def input_():
    while True:
        try:
            N = int(input("Введите N: "))
            k = int(input("Введите K: "))
            if not (1 < k < N):
```

```
        print("Ошибка: K должно быть больше 1 и меньше N")
        continue
    lst = [random.randint(-100, 100) for _ in range(N)]
    new_lst = [0]*k + lst[:-k]
    return lst, new_lst
except ValueError:
    print("Ошибка: введите число")

print(input_())
```

Вывод: В процессе работы я закрепил усвоенные знания, понятия, научился работать со строками.