

Тема: Составление программ с функциями в IDE PyCharm Community.

Цель: Даны пелыс числа N (2), A и B . Сформировать и вывести полочисленный список размера 10, первый элемент которого равен A , второй равен B , а каждый последующий элемент равен сумме всех предыдущих. python

Постановка задачи: Опишите функцию `calculate_sum()`, которая находит сумму чисел в последовательности от 1 до 60, используя локальные переменные. Далее, вызовите эту функцию для нахождения суммы и выведите результат.

Задача 1

Постановка задачи.

Дан целочисленный список размера N . Увеличить все нечетные числа, содержащиеся в списке, на исходное значение последнего нечетного числа. Если нечетные числа в списке отсутствуют, то оставить список без изменений.

Текст программы:

```
# Вариант 6
# Даны пелыс числа N (2), A и B.
# Сформировать и вывести
# полочисленный список размера 10,
# первый элемент которого равен A,
# второй равен B, а каждый
# последующий элемент равен сумме
# всех предыдущих. python
def generate_sequence(a, b, n):
    sequence = [a, b] #
    инициализируем список первыми
    двумя числами

    for i in range(2, n):
        next_number = sequence[i
- 1] + sequence[i - 2] #
    вычисляем следующее число как
    сумму двух предыдущих

    sequence.append(next_number) #
    добавляем вычисленное число в
    список

    return sequence # возвращаем
    полученный список

# Вводим значения
a = int(input("Введите значение
A: "))
b = int(input("Введите значение
B: "))
n = 10 # размер списка
```

Протокол работы программы:

Введите значение A: 12

Введите значение B: 16

[12, 16, 28, 44, 72, 116, 188, 304, 492, 796]

Process finished with exit code 0

Process finished with exit code 0

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

Задание 2

Постановка задачи:

Дан список размера N. Найти максимальный из его локальных минимумов
(локальный минимум -

- это элемент, который меньше любого из своих соседей).

00:07

Текст
программы:

```
#Вариант 6
#Дан список размера N. Найти
максимальный из его локальных
минимумов (локальный минимум – это
элемент, который меньше любого из
своих соседей).
def find_local_minimums(arr):
    local_minimums = []

    for i in range(1, len(arr) -
1):
        if arr[i] < arr[i - 1] and
arr[i] < arr[i + 1]:

local_minimums.append(arr[i])

    return local_minimums

def find_max_local_minimum(arr):
    local_minimums =
find_local_minimums(arr)

    if len(local_minimums) == 0:
        return None # если
локальных минимумов нет, вернуть
None
    else:
        return max(local_minimums)
# вернуть наибольший из локальных
минимумов

# Пример использования
input_list = [3, 1, 4, 1, 5, 9, 2,
6, 5, 3, 5]
result =
find_max_local_minimum(input_list)
print("Наибольший из локальных
минимумов:", result)
```

Протокол работы программы:]

Наибольший из локальных минимумов: 3

Process finished with exit code 0

Process finished with exit code 0

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

Задача 3

Постановка задачи.

#Дан список размера N, все элементы которого, кроме последнего, упорядочены по возрастанию. #Сделать список упорядоченным, переместив последний элемент на новую позицию.

Текст программы:

```
#Вариант 6
#Дан список размера N и целое
число K (1 < K < N). Осуществить
сдвиг элементов список вправо на
K позиций (при этом A 1 перейдет
в Akn, A2 — в Ak+2, .AN-k - An, а
исходное значение K последних
элементов будет потеряно). Первые
K элементов полученного списка
положить равными 0.
def
shift_and_replace_with_zeros(arr,
k):
    n = len(arr)
    k %= n # если K > N, мы
вычисляем эффективное значение k

    # Создаем новый список с
нужными смещениями
    shifted_list = arr[-k:] +
arr[:-k]
    # Заменяем первые K элементов
полученного списка нулями
    for i in range(k):
        shifted_list[i] = 0

    return shifted_list

# Пример использования
original_list = [1, 2, 3, 4, 5,
6, 7, 8, 9]
K = 3
result =
shift_and_replace_with_zeros(orig
inal_list, K)
print(result)
```

Протокол работы программы:

[0, 0, 0, 1, 2, 3, 4, 5, 6]

Текст
программы:

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