

Задача №10

Написать функцию, которая принимает положительное целое число n и определяющая является ли число n сбалансированным. Число является сбалансированным, если сумма цифр до средних цифр равна сумме цифр после средней цифры. Если число нечетное (длина числа), то средняя цифра одна, если четное, то средних цифр две. При расчете, средние числа не участвуют.

Например:

- Число 23441 сбалансированное, так как $2+3=4+1$
- Число 7 сбалансированное, так как $0=0$
- Число 1231 сбалансированное, так как $1=1$
- Число 123456 несбалансированное, так как $1+2 \neq 5+6$

Решение:

При считывании проверим, что действительно получили положительное целое число.

Для удобства работы сохраним число в виде строки (возможность обращаться по индексам, делать срезы, определять количество чисел, из которых состоит число)

Рассмотрим 3 ситуации:

1) Если строка имеет длину равную 1 или 2, то в таком случае суммы цифр до средних чисел равны 0, можно выводить результат --> сбалансированное число.

2) Иначе сделаем два среза строки lft и rgt: соответственно подстрока из левых цифр относительно середины и правых.

$\text{numb} = \overset{0}{1} \overset{1}{2} \overset{2}{3} \overset{3}{4} \overset{4}{5}$ lft = numb [:len(numb)//2]

rgt = numb[len(numb)//2+1:]

$\text{numb} = \overset{0}{1} \overset{1}{2} \overset{2}{3} \overset{3}{4} \overset{4}{5} \overset{5}{6}$ lft = numb [:len(numb)//2 - 1]

Сравним левый и правый срез - если состоят из одинаковых символов, значит в сумме дают одни и те же значения.

3) Иначе в цикле, проходящем по всем элементам срезов, прибавляем в переменной flag числа из левого среза и вычитаем значения из правого среза - если в итоге получим 0, то сумма чисел левого среза равна сумме чисел правого среза, то есть число является сбалансированным.

Тестирование:

```
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
hhjhdfjs
Введен некорректный параметр. Попробуйте снова.
Введите положительное целое число.
-23323
Введен некорректный параметр. Попробуйте снова.
Введите положительное целое число.
0
Введен некорректный параметр. Попробуйте снова.
Введите положительное целое число.
7
Число 7 сбалансированное, так как 0 = 0.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
23441
Число 23441 сбалансированное, так как 2 + 3 = 4 + 1.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
1231
Число 1231 сбалансированное, так как 1 = 1.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
123456
Число 123456 несбалансированное, так как 1 + 2 != 5 + 6.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
15
Число 15 сбалансированное, так как 0 = 0.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
242
Число 242 сбалансированное, так как 2 = 2.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
2567
Число 2567 несбалансированное, так как 2 != 7.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
16807
Число 16807 сбалансированное, так как 1 + 6 = 0 + 7.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
59873
Число 59873 несбалансированное, так как 5 + 9 != 7 + 3.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % python3 balance.py
Введите положительное целое число.
1234563000777
Число 1234563000777 сбалансированное, так как 1 + 2 + 3 + 4 + 5 + 6 = 0 + 0 + 0 + 7 + 7 + 7.
[anastasiia_maxim@MacBook-Pro-Anastasiia pythonProject1 % ]
```