

Задача 2. Сумма чётных 2

Источник:	базовая*
Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

Требуется реализовать функцию `foreach`, которая пробегает по всем элементам заданного **массива** целых чисел и вызывает указанный `callback` для каждого из этих элементов. Аналогичную функцию `foreach` нужно реализовать и для **связного списка**. Передаваемый `callback` должен поддерживать контекст.

Пример сигнатуры функции `foreach` для массива приведён ниже. Для связного списка реализуйте функцию `foreach` с аналогичной сигнатурой, чтобы можно было передавать в неё те же самые `callback`-функции.

```
//тип указателя на функцию, которую можно передавать как callback
typedef void (*callback)(void *ctx, int *value);
//здесь ctx -- это контекст, который передаётся в func первым аргументом
//последние два параметра задают массив, по элементам которого нужно пройтись
void arrayForeach(void *ctx, callback func, int *arr, int n);
```

Эти две функции требуется применить для решения следующей тестовой задачи: дана последовательность целых чисел, требуется посчитать сумму всех её чётных элементов. Запишите данную последовательность как в массив, так и в связный список, и вызовите в каждом случае `foreach`. Функция-`callback`, которую вы передаёте в `foreach`, не должна обращаться к глобальным переменным: все необходимые для её работы данные передавайте через контекст.

Формат входных данных

В первой строке содержится целое число N — количество элементов последовательности ($1 \leq N \leq 100$). Во второй строке записано N целых чисел через пробел — сама последовательность. Все элементы последовательности по абсолютной величине не превышают 100.

Формат выходных данных

Выведите два целых числа: сумму всех чётных элементов последовательности. Первое число должно соответствовать сумме, вычисленной с помощью `foreach` по массиву, а второе — сумме, вычисленной с помощью `foreach` по связному списку.

Пример

input.txt	output.txt
8 2 3 7 6 8 3 1 2	18 18