

## Задание 1

Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 ... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число  $n$  — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.

Например, если  $n = 7$ , то программа должна вывести 1 2 2 3 3 3 4.

---

### Sample Input:

7

---

### Sample Output:

1 2 2 3 3 3 4

## Задание 2

Напишите программу, на вход которой подаётся прямоугольная матрица в виде последовательности строк. После последней строки матрицы идёт строка, содержащая только строку "end" (без кавычек, см. Sample Input).

Программа должна вывести матрицу того же размера, у которой каждый элемент в позиции  $i, j$  равен сумме элементов первой матрицы на позициях  $(i-1, j)$ ,  $(i+1, j)$ ,  $(i, j-1)$ ,  $(i, j+1)$ . У крайних символов соседний элемент находится с противоположной стороны матрицы.

В случае одной строки/столбца элемент сам себе является соседом по соответствующему направлению.

---

### Sample Input 1:

9 5 3  
0 7 -1  
-5 2 9  
end

---

### Sample Output 1:

3 21 22  
10 6 19  
20 16 -1

---

### Sample Input 2:

1  
end

---

### Sample Output 2:

4

### Задание 3

Напишите программу, на вход которой подаётся список чисел одной строкой. Программа должна для каждого элемента этого списка вывести сумму двух его соседей. Для элементов списка, являющихся крайними, одним из соседей считается элемент, находящийся на противоположном конце этого списка. Например, если на вход подаётся список "1 3 5 6 10", то на выход ожидается список "13 6 9 15 7" (без кавычек).

Если на вход пришло только одно число, надо вывести его же.

Вывод должен содержать одну строку с числами нового списка, разделёнными пробелом.

---

**Sample Input 1:**

1 3 5 6 10

---

**Sample Output 1:**

13 6 9 15 7

---

**Sample Input 2:**

10

---

**Sample Output 2:**

10

### Задание 4

Узнав, что ДНК не является случайной строкой, только что поступившие в Институт биоинформатики студенты группы информатиков предложили использовать алгоритм сжатия, который сжимает повторяющиеся символы в строке.

Кодирование осуществляется следующим образом:

`s = 'aaaabbcaa'` преобразуется в `'a4b2c1a2'`, то есть группы одинаковых символов исходной строки заменяются на этот символ и количество его повторений в этой позиции строки.

Напишите программу, которая считывает строку, кодирует её предложенным алгоритмом и выводит закодированную последовательность на стандартный вывод. Кодирование должно учитывать регистр символов.

---

**Sample Input 1:**

aaaabbcaa

---

**Sample Output 1:**

a4b2c1a2

---

**Sample Input 2:**

abc

---

**Sample Output 2:**

a1b1c1

**Задание 5**

Напишите функцию `update_dictionary(d, key, value)`, которая принимает на вход словарь `dd` и два числа: `key` и `value`.

Если ключ `key` есть в словаре `dd`, то добавьте значение `value` в список, который хранится по этому ключу.

Если ключа `key` нет в словаре, то нужно добавить значение в список по ключу `2 * key`. Если и ключа `2 * key` нет, то нужно добавить ключ `2 * key` в словарь и сопоставить ему список из переданного элемента `[value]`.

Требуется реализовать только эту функцию, кода вне её не должно быть.

Функция не должна вызывать внутри себя функции `input` и `print`.

Пример работы функции:

```
d = {}  
print(update_dictionary(d, 1, -1)) # None  
print(d) # {2: [-1]}  
update_dictionary(d, 2, -2)  
print(d) # {2: [-1, -2]}  
update_dictionary(d, 1, -3)  
print(d) # {2: [-1, -2, -3]}
```

**Задание 6**

Напишите программу, которая считывает из файла строку, соответствующую тексту, сжатому с помощью кодирования повторов, и производит обратную операцию, получая исходный текст.

Запишите полученный текст в файл и прикрепите его, как ответ на это задание.

В исходном тексте не встречаются цифры, так что код однозначно интерпретируем.

---

**Sample Input:**

a3b4c2e10b1

---

**Sample Output:**

aaabbbbcceeeeeeeeeeb