

# NSU-2023-T03L1e03

Для 8-битного *беззнакового* числа  $N$ , размещенного по адресу  $n$ , используйте цикл `while` чтобы вычислить сумму всех беззнаковых чисел между 0 и  $N$ .

- Если вычисление завершилось **без беззнакового переполнения**, ваша программа должна сохранить сумму по адресу памяти `res` и целочисленное значение  $(-1)$  в следующей ячейке памяти, помеченной `oflow`.
- Если вычисление **приводит к переполнению** в любой момент во время суммирования ваша программа должна сохранить значение 255 по адресу `res` и целое число 1 по адресу `oflow`.

Алгоритм выглядит следующим образом:

```
num = 0
total = 0
ov = -1
while (num <= N)
{
    total = total + num
    if an overflow occurs
    {
        ov = +1
        total = 255
        break
    }
    num = num + 1
}
store total
store ov
```

Проверьте свою программу, размещая разные беззнаковые числа по адресу  $n$

Заполните шаблон решения, предоставленный в `NSU-2023-T03L1e03.asm`,

в котором мы объявили метки  $n$ ,  $res$  и  $oflow$ .

## Соглашение о вызовах

Оставьте адрес результата в **r0**. (Это уже закодировано в шаблоне).