

# NSU-2023-T01L2e02

Напишите программу, которая складывает два 16-разрядных беззнаковых числа,  $A$  и  $B$ , расположенные в памяти в прямом (little-endian, от младшего к старшему) порядке, и сохраните результат как 16-разрядное (2-х байтовое) число, также в порядке от младшего к старшему.

Напоминание: 2-байтовое число, сохраненное в порядке от младшего к старшему начиная с адреса памяти  $X$ , должно иметь младший байт (правую сторону в арабской позиционной записи) по адресу  $X$  и старший байт (левую сторону) по адресу  $X + 1$ .

Начальные данные вашей программы должны быть размещены в памяти следующим образом:

| location | label      | description      | type            |
|----------|------------|------------------|-----------------|
| INPUTS+0 | <i>aLo</i> | младший байт $A$ | 8-битная строка |
| INPUTS+1 | <i>aHi</i> | старший байт $A$ | 8-битная строка |
| INPUTS+2 | <i>bLo</i> | младший байт $B$ | 8-битная строка |
| INPUTS+3 | <i>bHi</i> | старший байт $B$ | 8-битная строка |

Проверьте свою программу, размещая разные пары двухбайтовых значений, размещая соответствующие битовые строки по адресам памяти *aLo*, *aHi*, *bLo* и *bHi*.

Заполните шаблон решения, предоставленный в `NSU-2023-T01L2e02.asm`, в котором мы объявили метки `aLo`, `aHi`, `bLo`, `bHi`, `resLo` и `resHi`.

## Соглашение о вызовах

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