

Практическое занятие «Ассемблер–4»

17 марта 2020 года

Ниже под словами «задан массив» подразумевается, что имеется переменная или константа, содержащая длину массива, и блок памяти, непосредственно содержащий набор данных.

1. Задан массив 16-битных чисел без знака. Найти их сумму и произведение и записать в **32-битные** переменные `sum` и `prod`. То есть *не гарантируется*, что сумма и произведение помещаются в 16-битные переменные.
2. В 1-байтовой переменной `n` задано беззнаковое число, не превосходящее 12. Найти факториалы натуральных чисел от 0 до этой величины и записать их в массив 4-байтных беззнаковых чисел `facts` подходящего размера.
3. В массиве 16-битных беззнаковых чисел `years` записаны номера лет. В результате работы программы массив 1-байтовых чисел `leaps` должен заполниться нулями и единицами: в i -й ячейке стоит нуль, если i -й элемент массива `years` содержит невисокосный год, 1 — если високосный. Напомним, что год является високосным, если его номер делится на 4 и не делится на 100 или если номер делится на 400.
4. Задан массив 16-битных чисел со знаком. Переработать их в массив 8-битных беззнаковых чисел, содержащих их последние цифры. Например,
$$1 \ -23 \ 15 \ 137 \rightarrow 1 \ 3 \ 5 \ 7.$$
5. Задан массив байтов. Перепишите его в другой массив в обратном порядке.
6. Задан массив 32-битных чисел со знаком. Перепишите в другой массив числа, кратные числу, заданному в переменной `k` (отличному от нуля). Длина результирующего массива должна быть сохранена в переменной!
- 7*. Задан массив 16-битных беззнаковых чисел. Также имеется достаточно большой массив байтов (общее количество битов в элементах этого массива превосходит длину первого массива). Требуется найти **32-битную** сумму тех элементов из первого массива, которые отвечают единичным **битами** в элементах второго массива: младший бит в первом элементе второго массива отвечает нулевому элементу первого, следующий бит в первом элементе второго массива отвечает первому элементу первого, старший (7-й) бит первого элемента второго массива отвечает 7-му элементу первого, младший бит второго элемента второго массива отвечает восьмому элементу первого, и т.д. Например,

```
ar1 dw 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
```

```
ar2 db 10010110B, 00000010B
```

Искомая сумма равна $2 + 3 + 5 + 8 + 10 = 28$.