

# Вариант № 19

## Задание №4

### Условие

Разработать ассемблер и интерпретатор для учебной виртуальной машины (УВМ). Система команд УВМ представлена далее.

Для ассемблера необходимо разработать читаемое представление команд УВМ. Ассемблер принимает на вход файл с текстом исходной программы, путь к которой задается из командной строки. Результатом работы ассемблера является бинарный файл в виде последовательности байт, путь к которому задается из командной строки. Дополнительный ключ командной строки задает путь к файлу-логу, в котором хранятся ассемблированные инструкции в духе списков “ключ=значение”, как в приведенных далее тестах.

Интерпретатор принимает на вход бинарный файл, выполняет команды УВМ и сохраняет в файле-результате значения из диапазона памяти УВМ. Диапазон также указывается из командной строки. Форматом для файла-лога и файла-результата является уaml. Необходимо реализовать приведенные тесты для всех команд, а также написать и отладить тестовую программу.

### Загрузка константы

A	B	C
Биты 0—2	Биты 3—6	Биты 7—34
7	Адрес	Константа

Размер команды: 5 байт. Операнд: поле C. Результат: регистр по адресу, которым является поле B.

Тест (A=7, B=2, C=519):

0x97, 0x03, 0x01, 0x00, 0x00

### Чтение значения из памяти

A	B	C
Биты 0—2	Биты 3—6	Биты 7—10
3	Адрес	Адрес

Размер команды: 5 байт. Операнд: значение в памяти по адресу, которым является регистр по адресу, которым является поле C. Результат: регистр по адресу, которым является поле B.

Тест (A=3, B=7, C=7):

0xBB, 0x03, 0x00, 0x00, 0x00

Запись значения в память

A	B	C	D
Биты 0—2	Биты 3—6	Биты 7—11	Биты 12-15
3	Адрес	Смещение	Адрес

Размер команды: 5 байт. Операнд: регистр по адресу, которым является поле D. Результат: значение в памяти по адресу, которым является сумма адреса (регистр по адресу, которым является поле B) и смещения (поле C).

Тест (A=1, B=9, C=13, D=15):

0xC9, 0xF6, 0x00, 0x00, 0x00

Унарная операция: bitreverse()

A	B	C	D
Биты 0—2	Биты 3—6	Биты 7—11	Биты 12-15
1	Адрес	Смещение	Адрес

Размер команды: 5 байт. Операнд: регистр по адресу, которым является поле D. Результат: значение в памяти по адресу, которым является сумма адреса (регистр по адресу, которым является поле B) и смещения (поле C).

Тест (A=4, B=4, C=28, D=1):

0x24, 0x1E, 0x00, 0x00, 0x00

Тестирование

Программа успешно прошла тестирование

Исходная программа

```
set a0 0
set a1 1
set a3 42

mov test a3
shift a0 15 a3

bitreverse test 5 a0
bitreverse test 6 a1
```

Логи ассемблера

```
- A: 7
  B: 0
  C: 0
  bin: 0xe0, 0x00, 0x00, 0x00, 0x00
- A: 7
  B: 1
  C: 1
  bin: 0xe2, 0x00, 0x00, 0x00, 0x20
- A: 7
  B: 2
  C: 42
  bin: 0xe4, 0x00, 0x00, 0x50, 0x40
- A: 3
  B: 3
  C: 2
  bin: 0x66, 0x40, 0x00, 0x00, 0x00
- A: 1
  B: 0
  C: 15
  D: 2
  bin: 0x20, 0xf2, 0x00, 0x00, 0x00
- A: 4
  B: 3
  C: 5
  D: 0
  bin: 0x86, 0x50, 0x00, 0x00, 0x00
- A: 4
  B: 3
  C: 6
  D: 1
  bin: 0x86, 0x61, 0x00, 0x00, 0x00
```

Результат (Дамп памяти)

```
- '0b0000': 0
- '0b0001': 1
- '0b0010': 42
- '0b0011': 42
- '0b0100': 0
- '0b0101': 0
- '0b0110': 0
- '0b0111': 0
- '0b1000': 0
- '0b1001': 134217728
- '0b1010': 0
- '0b1011': 0
- '0b1100': 0
- '0b1101': 0
- '0b1110': 0
- '0b1111': 42
```