

Лабораторная работа №6.

Поддержка сегментной модели памяти в NASM и исполняемые файлы формата MZ

Макаров П. А.

1 Краткая теория

1. Netwide Assembler webpage;
2. Перевод руководства, выполненный AsmOs group и доступный на www.opennet.ru;
3. NASM — Википедия;
4. x86 memory models — Wikipedia;
5. x86 memory segmentation — Wikipedia;
6. DOS MZ executable — Wikipedia;

2 Задание для самостоятельного решения

1. Напишем и исследуем первое простейшее 16-разрядное DOS-приложение с поддержкой сегментной модели памяти. Исходный код приведён ниже.

```
segment data
hello: db 'Hello, world!', 0x0D, 0x0A, '$'
```

```
segment code
..start:
    mov ax, data
    mov ds, ax
```

```
print:
```

```

mov dx, hello
mov ah, 09
int 0x21

exit:
mov ax, 0x4C00
int 0x21

segment stack class=stack
resb 256

```

Транслировать и скомпоновать программу можно следующим образом.

```

C:\>nasm -Wall -f obj firstexe.asm -o first.obj
C:\>wlink file test.obj format dos name first.exe

```

После этого можно убедиться, что программа работает именно так, как и было задумано.

```

C:\>first.exe

```

А теперь можно исследовать её детально с помощью отладчика.

```

C:\>debug
-n first.exe
-l
-r          ; Внимательно посмотрите на сегментные регистры, IP и SP
-u 10 20    ; Обратите внимание на значение, загружаемое в регистр DX
-d 100 120  ; Сравните вывод данной и предыдущей команд.
              ; Сколько сегментов памяти вы здесь видите?
-d 100 220  ; А так?
-p
-p
-p
-p
-p
-p
-p

```

И в завершении этого исследования посмотрите содержимое файла `first.exe` в шестнадцатеричных кодах с помощью встроенного просмотрщика файлового менеджера Norton Commander (C:\>nc\nc, выделение курсором нужного файла, а затем клавиши F3 и F4).

2. Напишите программу, вычисляющую одну арифметическую операцию над двумя операндами, введёнными пользователем с клавиатуры. Программа должна иметь понятный интерфейс. Предусмотрите все возможные ошибки пользователя и продумайте реакцию программы на них. У каждого студента свой вариант задачи, который кодируется строкой из пяти символов: **AOPNS**, где **AOP** — арифметическая операция (**add**, **sub**, **mul**, **div**), **N** — количество разрядов операндов, **S** — символ системы счисления.