Лабораторная работа №4. Простейшие программы на ассемблере. Системные прерывания. Символьный ввод-вывод. Циклы. Строки и буферы.

Макаров П. А.

1 Краткая теория

1. x86 Assembly Language — Википедия;
2. R. Toal. x86 Assembly Language Programming;
3. A. Ferrari. A Tiny Guide to Programming in 32-bit x86 Assembly Language;
4. D. Evans. x86 Assembly Guide;
5. `http://www.techhelpmanual.com`
6. DOS API — Википедия;

2 Задания для самостоятельного решения

В первую очередь необходимо включить DOSBox и запустить в нём отладчик `DEBUG`.

```
C:\>debugers\debug\debug
```

1. Ещё раз об исполняемых файлах формата `COM`, PSP и инструкции `ret`.

```
-r
-d fffe
-u 0 1
-p=100
-p
```

Что здесь происходит? Как здесь задействован стек? Какую роль играет начало сегмента памяти длиной 255 байт? Каков максимальный размер программы формата COM и почему? Что происходит при выполнении инструкции int 20?

2.
```
-a
:0100 mov ah, 02
:0102 mov dl, 30
:0104 int 21
:0106 mov ax, 4c00
:0109 int 21
:010B
-r cx b
-n dsplzero.com
-w
-q
C:\>DIR
C:\>DSPLZERO
```

3.
```
C:\>debugers\debug\debug
-n dsplzero.com
-l
-u 100 10a
-d 100 10a
-e 103
:0103  30.39  CD.
-e 107
:0107  00.21  4C.
-g=100
-d 100 10a
```

4.
```
-a
:0100 mov ah, 01
:0102 int 21
:0104
-p=100
-p
```

5. Напишите программу, печатающую на экране символы 0123456789.

```
mov cx, 000A
mov ah, 02
```

```
mov dl, 30
int 21
inc dl
loop 107
mov ax, 4C00
int 21
```

6. Теперь реализуйте вывод тех же символов в обратном порядке.

7. Напишите программу, иллюстрирующую принцип действия функции `09h` системного прерывания ОС DOS.

8. По аналогии с предыдущей задачей, изучите на практике действие функции DOS `0Ah`.