

Отчёт по лабораторной работе №8

Межеловский Александр Игоревич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выполнение самостоятельной работы	13
6	Выводы	15

Список иллюстраций

3.1	Арифметические операции в <code>nasm</code>	7
3.2	Команды умножения в <code>nasm</code>	8
4.1	Создание директории	9
4.2	Переход в директорию	10
4.3	Создание нового файла	11
4.4	Окно исходного кода программы	12
5.1	Компиляция программы	13
5.2	Компиляция и запуск программы	14

Список таблиц

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Задание

1. Ознакомление с арифметическими операциями
2. Ознакомление с остальными операциями
3. Символьные и численные данные в NASM

3 Теоретическое введение

6.2.2. Арифметические операции в NASM

6.2.2.1. Целочисленное сложение `add`.

Схема команды целочисленного сложения `add` (от англ. *addition* - добавление) выполняет сложение двух операндов и записывает результат по адресу первого операнда. Команда `add` работает как с числами со знаком, так и без знака и выглядит следующим образом:

```
add <операнд_1>, <операнд_2>
```

Допустимые сочетания операндов для команды `add` аналогичны сочетаниям операндов для команды `mov`.

Так, например, команда `add eax, ebx` прибавит значение из регистра `ebx` к значению из регистра `eax` и запишет результат в регистр `eax`.

Примеры:

```
add ax, 5      ; AX = AX + 5
add dx, cx     ; DX = DX + CX
add dx, cl     ; Ошибка: разный размер операндов.
```

Рис. 3.1: Арифметические операции в `nasm`

6.2.2.5. Команды умножения `mul` и `imul`.

Умножение и деление, в отличие от сложения и вычитания, для знаковых и беззнаковых чисел производятся по-разному, поэтому существуют различные команды.

Для *беззнакового* умножения используется команда `mul` (от англ. *multiply* – умножение):

`mul` <операнд>

Для *знакового* умножения используется команда `imul`:

`imul` <операнд>

Для команд умножения один из сомножителей указывается в команде и должен находиться в регистре или в памяти, но не может быть непосредственным операндом. Вторым сомножителем в команде явно не указывается и должен находиться в регистре `EAX`, `AX` или `AL`, а результат помещается в регистры `EDX:EAX`, `DX:AX` или `AX`, в зависимости от размера операнда 6.1.

Таблица 6.1. Регистры используемые командами умножения в Nasm

Размер операнда	Неявный множитель	Результат умножения
1 байт	AL	AX
2 байта	AX	DX:AX
4 байта	EAX	EDX:EAX

Рис. 3.2: Команды умножения в nasm

4 Выполнение лабораторной работы

Создаю директорию.

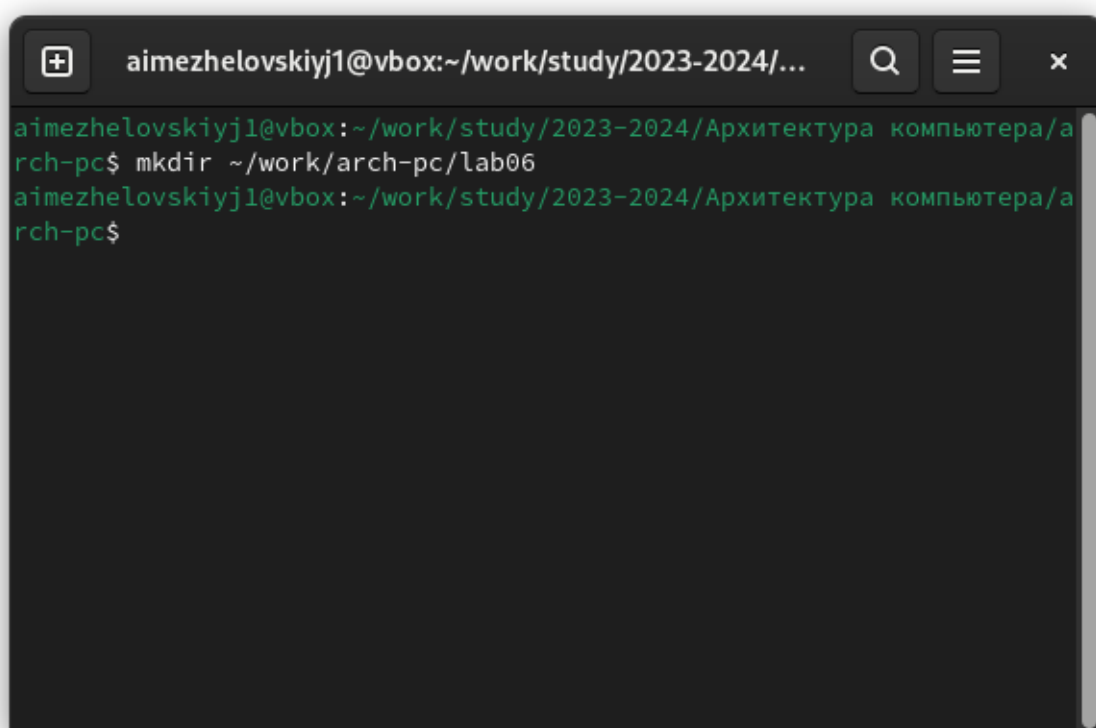
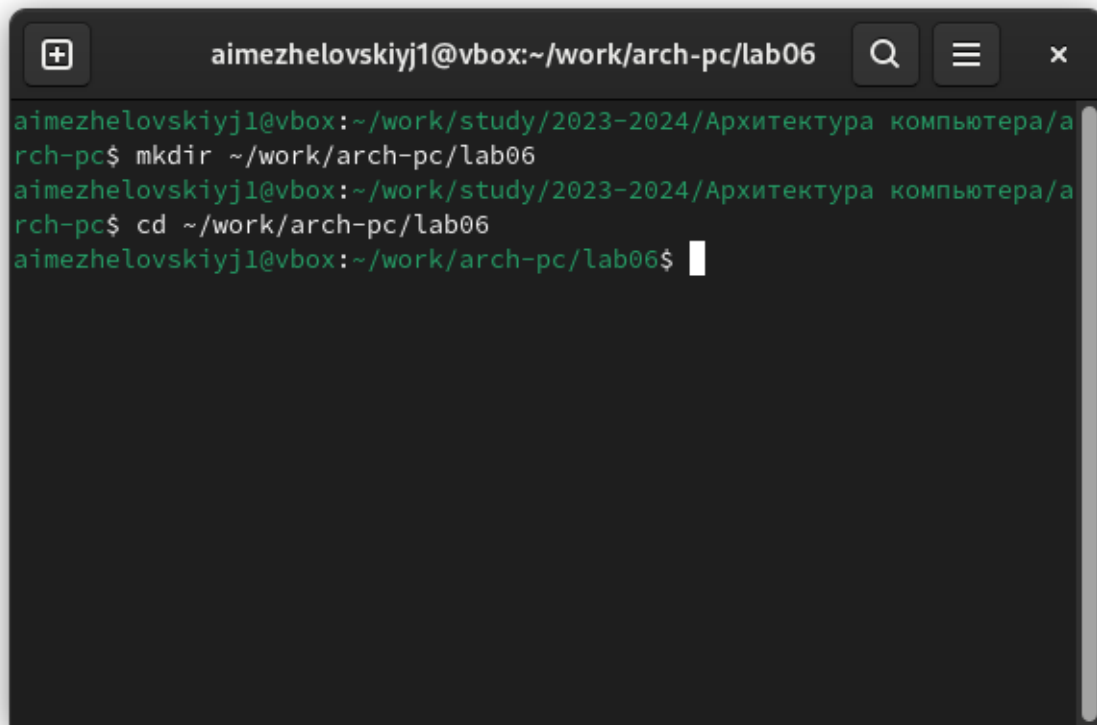
A screenshot of a terminal window with a dark background. The title bar at the top shows the username 'aimezhelovskiyj1@vbox' and the current directory path '~ /work/study/2023-2024/...'. There are search, menu, and close icons on the right. The terminal text shows the user 'aimezhelovskiyj1@vbox' at the prompt 'rch-pc\$' entering the command 'mkdir ~/work/arch-pc/lab06'. The prompt then changes to 'aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc\$'.

Рис. 4.1: Создание директории

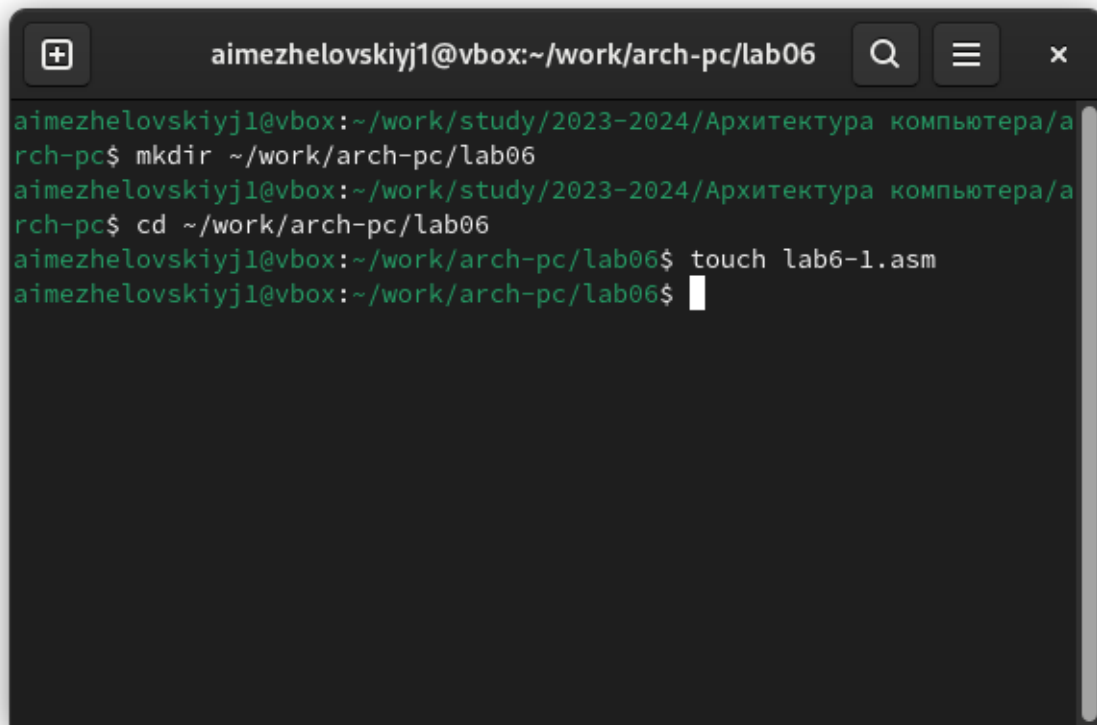
Перехожу в директорию.

A terminal window with a dark background and light green text. The window title bar shows the username 'aimezhelovskiyj1@vbox' and the current directory '~/work/arch-pc/lab06'. The terminal content shows a sequence of commands: first, 'mkdir ~/work/arch-pc/lab06' is executed; then, 'cd ~/work/arch-pc/lab06' is entered, and the prompt changes to reflect the new directory. The final line shows the prompt 'aimezhelovskiyj1@vbox:~/work/arch-pc/lab06\$' with a cursor.

```
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/a
rch-pc$ mkdir ~/work/arch-pc/lab06
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/a
rch-pc$ cd ~/work/arch-pc/lab06
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$
```

Рис. 4.2: Переход в директорию

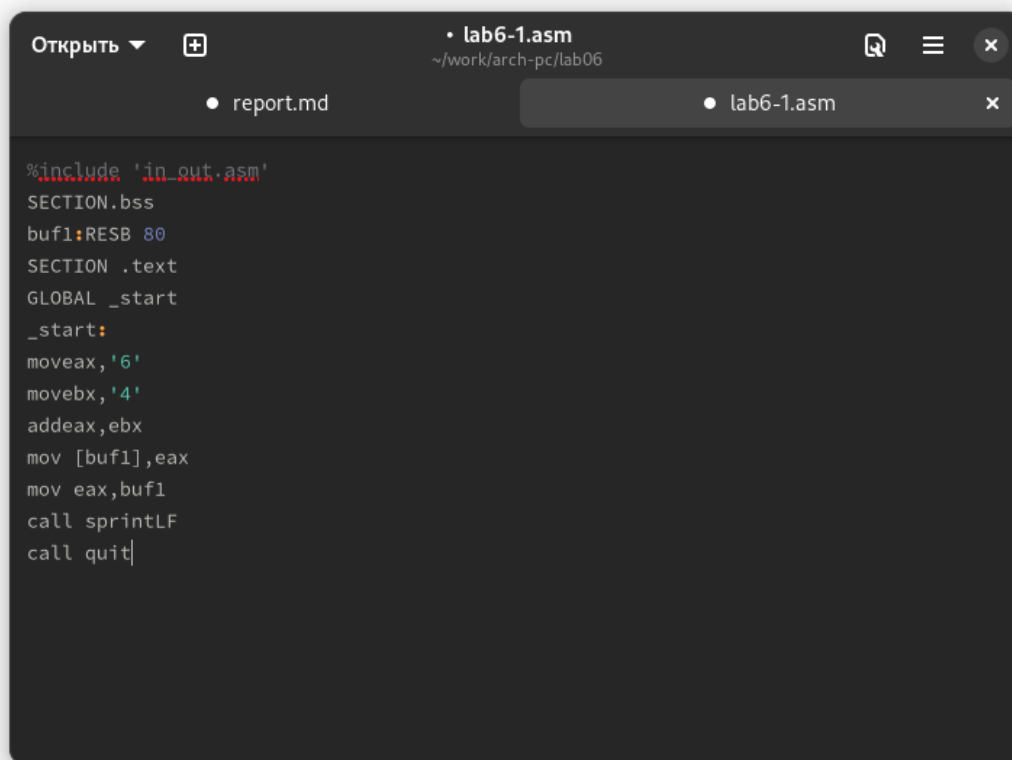
Создаю новый файл.

A terminal window with a dark background and light green text. The window title bar shows the user 'aimezhelovskiyj1@vbox' and the current directory '~/work/arch-pc/lab06'. The terminal contains the following commands and their outputs:

```
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc$ mkdir ~/work/arch-pc/lab06
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc$ cd ~/work/arch-pc/lab06
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ touch lab6-1.asm
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$
```

Рис. 4.3: Создание нового файла

Набираю исходный код программы в редакторе.



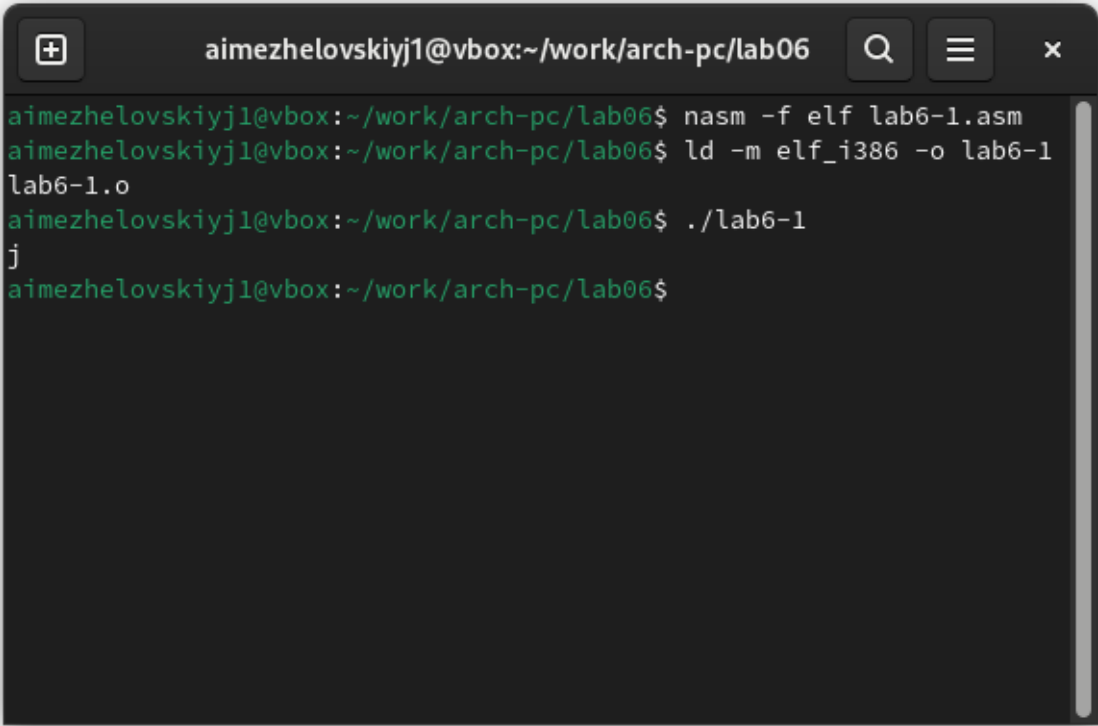
The image shows a code editor window with a dark theme. The title bar at the top contains the text "Открыть" (Open) with a dropdown arrow, a plus icon, and the file name "lab6-1.asm" with its path "~/.work/arch-pc/lab06". Below the title bar is a tab bar with two tabs: "report.md" and "lab6-1.asm". The main area of the window displays assembly code for "lab6-1.asm".

```
%include 'in_out.asm'
SECTION.bss
buf1:RESB 80
SECTION .text
GLOBAL _start
_start:
moveax,'6'
movebx,'4'
addeax,ebx
mov [buf1],eax
mov eax,buf1
call sprintf
call quit|
```

Рис. 4.4: Окно исходного кода программы

5 Выполнение самостоятельной работы

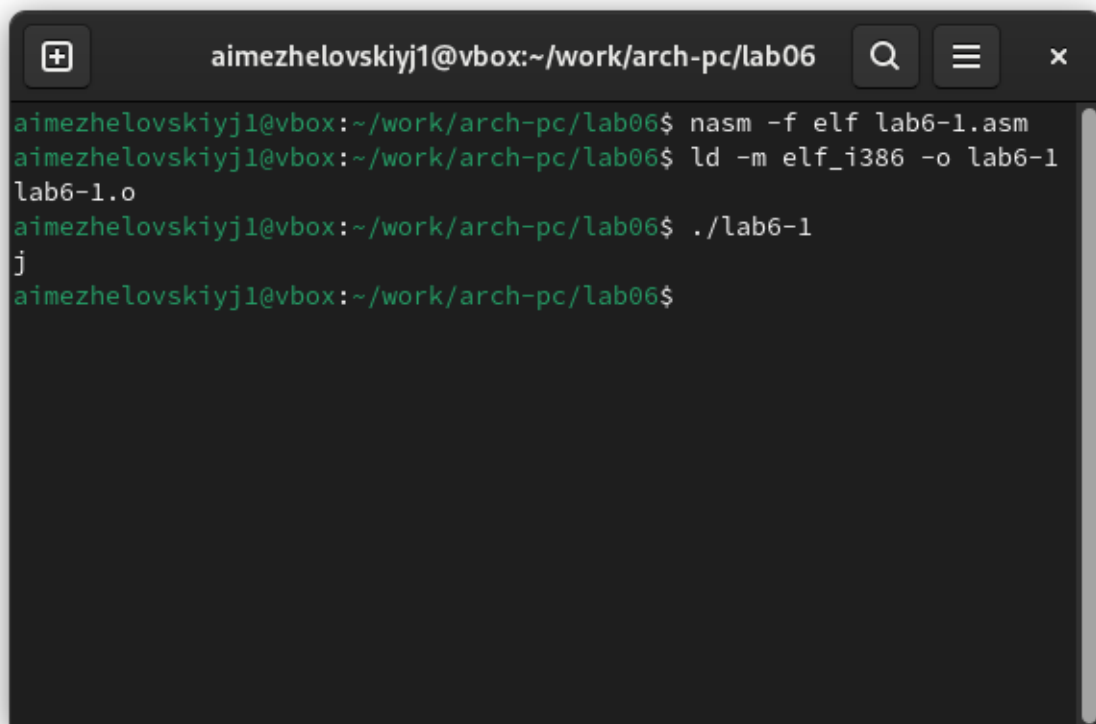
Компилирую программу.



```
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1
lab6-1.o
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ ./lab6-1
j
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$
```

Рис. 5.1: Компиляция программы

Компилирую программу с использованием функций, получаю желаемый результат.

A terminal window with a dark background and light green text. The window title is 'aimezhelovskiyj1@vbox:~/work/arch-pc/lab06'. The terminal shows the following commands and output:

```
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1
lab6-1.o
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$ ./lab6-1
j
aimezhelovskiyj1@vbox:~/work/arch-pc/lab06$
```

Рис. 5.2: Компиляция и запуск программы

6 Выводы

В результате проделанной работы я освоил арифметические инструкции языка ассемблера NASM.