

Шаблон отчёта по лабораторной работе №7

**Команды безусловного и условного переходов в Nasm.
Программирование ветвлений.**

Гомес Лопес Теофания

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Реализация переходов в NASM	7
3.2	Изучение структуры файлы листинга	11
3.3	Задание для самостоятельной работы	14
4	Выводы	18
	Список литературы	19

Список иллюстраций

3.1	Создала каталог с помощью команды <code>mkdir</code> и файл с помощью команды <code>touch</code>	7
3.2	Заполняла файл	8
3.3	Запускала файл и смотрим на его работу	8
3.4	Изменяла файл	9
3.5	Запускала файл и смотрела на его работу	9
3.6	Редактировала файл	10
3.7	Проверяла, сошелся ли наш вывод с данным в условии выводом . .	10
3.8	Создала файл с командой <code>touch</code>	10
3.9	Заполняла файл	11
3.10	Смотрела на работу программ	11
3.11	Создала файл листинга	12
3.12	Изучала файл	12
3.13	Удаляла операндум из файла	13
3.14	Транслировала файл	13
3.15	Изучала файл с ошибкой	14
3.16	Создала файл командой <code>touch</code>	14
3.17	Писала программу	15
3.18	Смотрела на работу программы	15
3.19	Создала файл командой <code>touch</code>	16
3.20	Писала программу	16
3.21	Проверяла работу программы	16
3.22	Проверяла работу программы	17

Список таблиц

1 Цель работы

Освоить условного и безусловного перехода. Ознакомиться с назначением и структурой файла листинга.

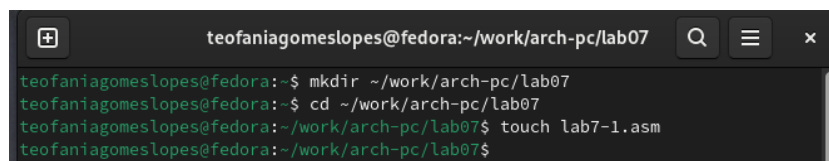
2 Задание

Написать программы для решения системы выражений.

3 Выполнение лабораторной работы

3.1 Реализация переходов в NASM

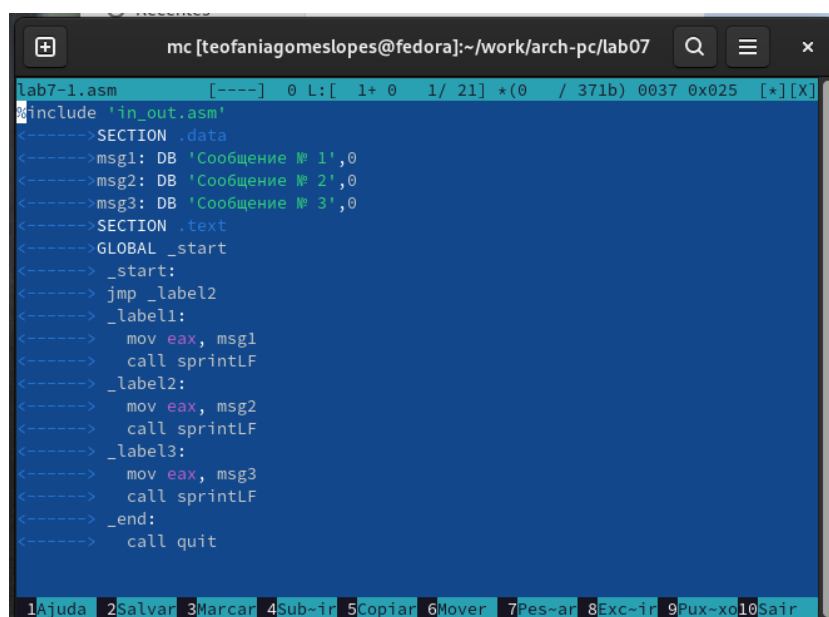
Создала каталог для программ ЛБ7, и в нем создала файл (рис. 3.1).



```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07
teofaniagomeslopes@fedora:~$ mkdir ~/work/arch-pc/lab07
teofaniagomeslopes@fedora:~$ cd ~/work/arch-pc/lab07
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ touch lab7-1.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.1: Создала каталог с помощью команды `mkdir` и файл с помощью команды `touch`

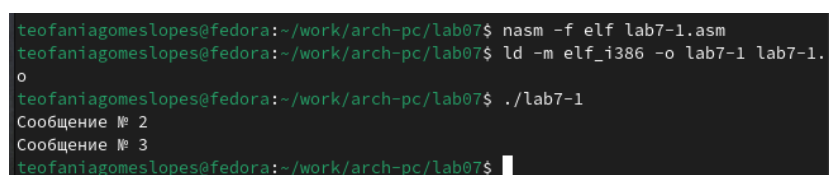
Открывала файл в Midnight Commander и заполняла его в соответствии с листингом 7.1 (рис. 3.2).



```
lab7-1.asm  [----]  0  L: [ 1+ 0 1/ 21] *(0 / 371b) 0037 0x025 [*][X]
%include 'in_out.asm'
<----->SECTION .data
<----->msg1: DB 'Сообщение № 1',0
<----->msg2: DB 'Сообщение № 2',0
<----->msg3: DB 'Сообщение № 3',0
<----->SECTION .text
<----->GLOBAL _start
<----->_start:
<-----> jmp _label2
<----->_label1:
<-----> mov eax, msg1
<-----> call sprintf
<----->_label2:
<-----> mov eax, msg2
<-----> call sprintf
<----->_label3:
<-----> mov eax, msg3
<-----> call sprintf
<----->_end:
<-----> call quit
```

Рис. 3.2: Заполняла файл

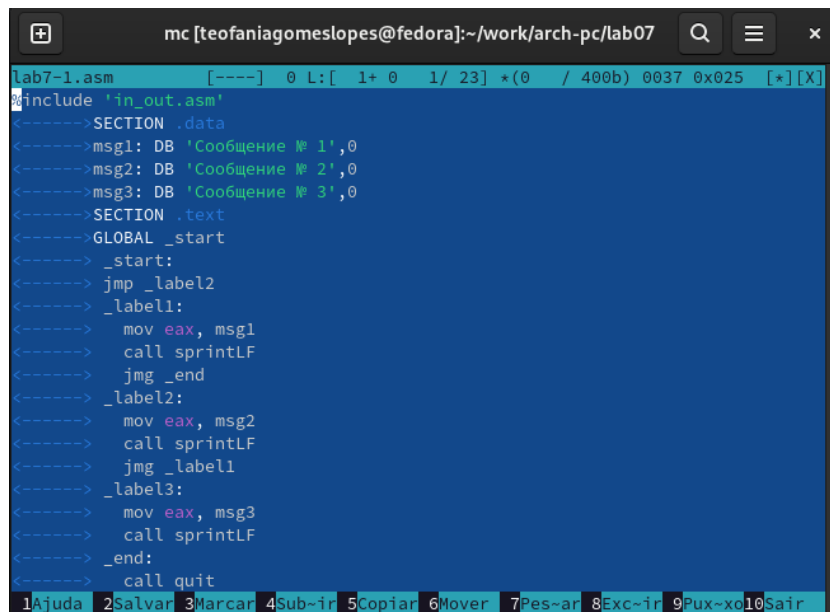
Создала исполняемый файл и запускаем его (рис. 3.3).



```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 3
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.3: Запускала файл и смотрим на его работу

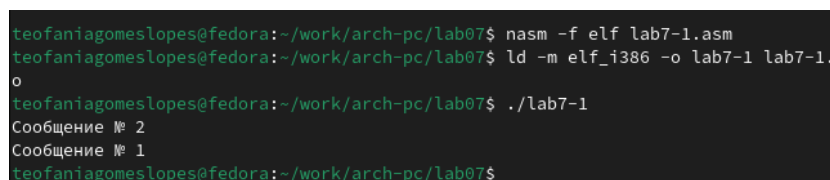
Снова открывала файл для редактирования и изменяла его, чтобы произошел данный вывод (рис. 3.4).



```
lab7-1.asm  [----] 0 L: [ 1+ 0 1/ 23] *(0 / 400b) 0037 0x025 [*][X]
%include 'in_out.asm'
<----->SECTION .data
<----->msg1: DB 'Сообщение № 1',0
<----->msg2: DB 'Сообщение № 2',0
<----->msg3: DB 'Сообщение № 3',0
<----->SECTION .text
<----->GLOBAL _start
<----->_start:
<-----> jmp _label2
<----->_label1:
<-----> mov eax, msg1
<-----> call sprintLF
<-----> jmp _end
<----->_label2:
<-----> mov eax, msg2
<-----> call sprintLF
<-----> jmp _label1
<----->_label3:
<-----> mov eax, msg3
<-----> call sprintLF
<----->_end:
<-----> call quit
```

Рис. 3.4: Изменяла файл

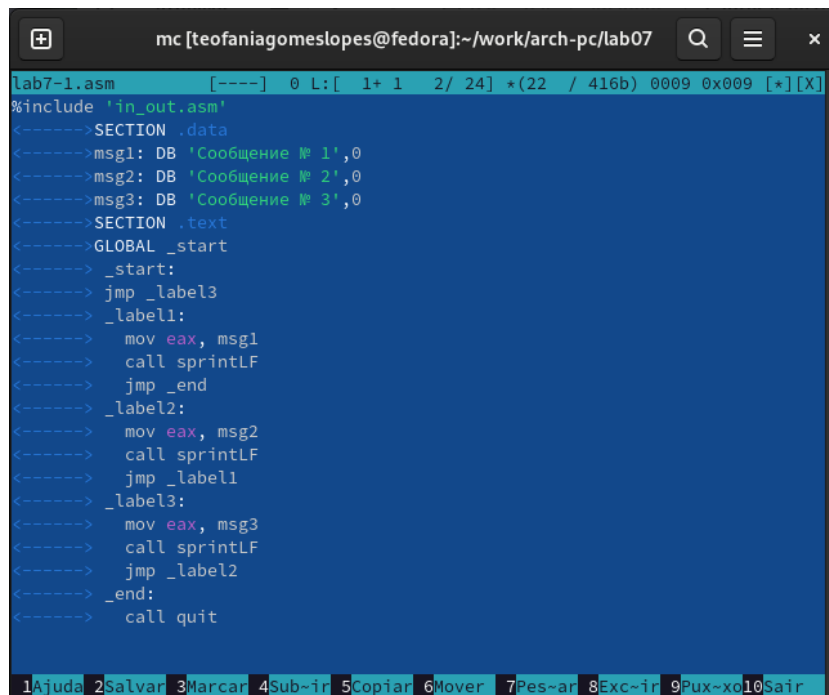
Создала исполняемый файл и запускала его (рис. 3.5).



```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 1
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.5: Запускала файл и смотрела на его работу

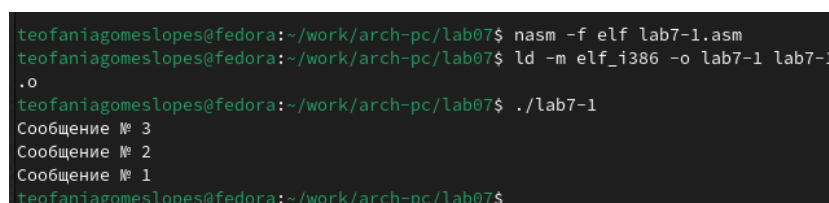
Снова открывала файл для редактирования и изменяла его, чтобы произошел данный вывод (рис. 3.6).



```
lab7-1.asm  [-----]  0 L: [ 1+ 1 2/ 24] *(22 / 416b) 0009 0x009 [*][X]
%include 'in_out.asm'
<----->SECTION .data
<----->msg1: DB 'Сообщение № 1',0
<----->msg2: DB 'Сообщение № 2',0
<----->msg3: DB 'Сообщение № 3',0
<----->SECTION .text
<----->GLOBAL _start
<----->_start:
<-----> jmp _label3
<----->_label1:
<-----> mov eax, msg1
<-----> call sprintf
<-----> jmp _end
<----->_label2:
<-----> mov eax, msg2
<-----> call sprintf
<-----> jmp _label1
<----->_label3:
<-----> mov eax, msg3
<-----> call sprintf
<-----> jmp _label2
<----->_end:
<-----> call quit
```

Рис. 3.6: Редактировала файл

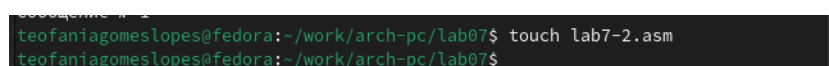
Создала исполняемый файл и запускала его (рис. 3.7).



```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 3
Сообщение № 2
Сообщение № 1
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.7: Проверяла, сошелся ли наш вывод с данным в условии выводом

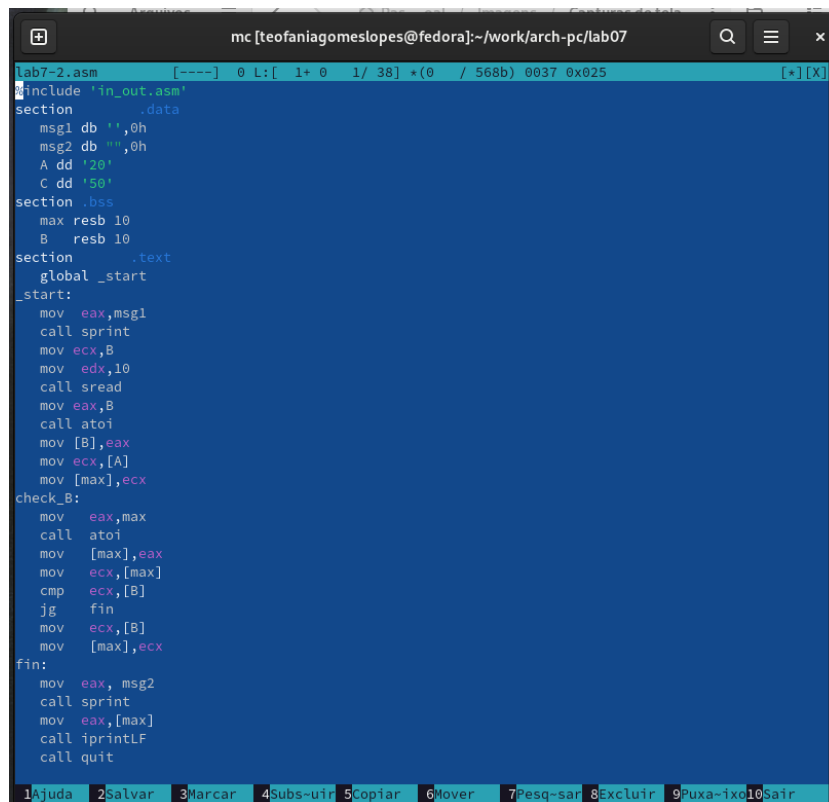
Создала новый файл (рис. 3.8).



```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ touch lab7-2.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.8: Создала файл с командой touch

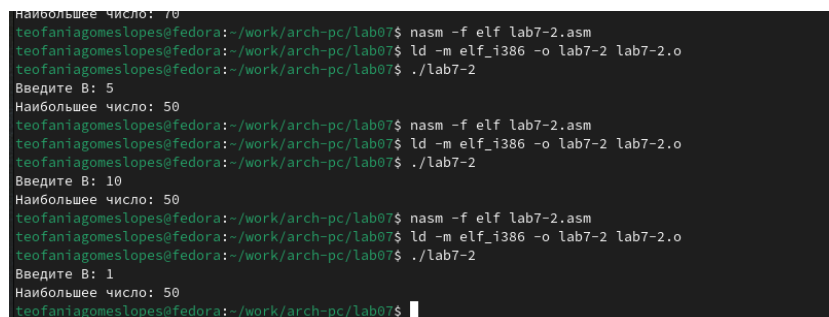
Открывала файл в Midnight Commander и заполняла его в соответствии с листингом 7.3 (рис. 3.9).



```
lab7-2.asm [----] 0 L: 1+ 0 1/ 38 *(0 / 568b) 0037 0x025 [*][X]
#include 'in_out.asm'
section .data
msg1 db '',0h
msg2 db "",0h
A dd '20'
C dd '50'
section .bss
max resb 10
B resb 10
section .text
global _start
_start:
mov eax,msg1
call sprint
mov ecx,B
mov edx,10
call sread
mov eax,B
call atoi
mov [B],eax
mov ecx,[A]
mov [max],ecx
check_B:
mov eax,max
call atoi
mov [max],eax
mov ecx,[max]
cmp ecx,[B]
jg fin
mov ecx,[B]
mov [max],ecx
fin:
mov eax, msg2
call sprint
mov eax,[max]
call iprintLF
call quit
```

Рис. 3.9: Заполняла файл

Создала исполняемый файл и проверяла его работу, вводя разные значения (рис. 3.10).



```
Наибольшее число: 70
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 5
Наибольшее число: 50
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 10
Наибольшее число: 50
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 1
Наибольшее число: 50
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.10: Смотрела на работу программ

3.2 Изучение структуры файлы листинга

Создала файл листинга для программы lab7-2.asm (рис. 3.11).

```

наибольшей длины: 50
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf -l lab7-2.lst lab7-2.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$

```

Рис. 3.11: Создала файл листинга

Открывала файл листинга с помощью команды `mcedit` и изучала его (рис. 3.12).

```

lab7-2.lst  [----]  0 L:[ 1+ 0 1/218] *(0 /13102b) 0032 0x020  [*][X]
1          %include 'in_out.asm'
2          <I> ;----- slen -----
3          <I> ; Функция вычисления длины сообщения
4          <I> slen:
5          00000000 53          <I> push    ebx
6          00000001 89C3       <I> mov     ebx, eax
7          <I> .....
8          <I> nextchar:
9          00000003 803800     <I> cmp     byte [eax], 0
10         00000006 7403       <I> jz      finished
11         00000008 40        <I> inc     eax
12         00000009 EBF8       <I> jmp     nextchar
13         <I> .....
14         <I> finished:
15         0000000B 29D8       <I> sub     eax, ebx
16         0000000D 5B        <I> pop     ebx
17         0000000E C3        <I> ret
18         <I> .....
19         <I> ;----- sprintf -----
20         <I> ; Функция печати сообщения
21         <I> ; входные данные: mov eax, <message>
22         <I> sprintf:
23         0000000F 52        <I> push    edx
24         00000010 51        <I> push    ecx
25         00000011 53        <I> push    ebx
26         00000012 50        <I> push    eax
27         00000013 E8E8FFFF    <I> call    slen
28         <I> .....
29         00000018 89C2       <I> mov     edx, eax
30         0000001A 58        <I> pop     eax
31         <I> .....
32         0000001B 89C1       <I> mov     ecx, eax
33         0000001D B801000000 <I> mov     ebx, 1
34         00000022 B804000000 <I> mov     eax, 4
35         00000027 CD80       <I> int     80h
36         <I> .....
37         00000029 5B        <I> pop     ebx
38         0000002A 59        <I> pop     ecx
39         0000002B 5A        <I> pop     edx
40         0000002C C3        <I> ret
41         <I> .....
42         <I> ;----- sprintfLF -----
43         <I> ; Функция печати сообщения с переводом строки
44         <I> .....

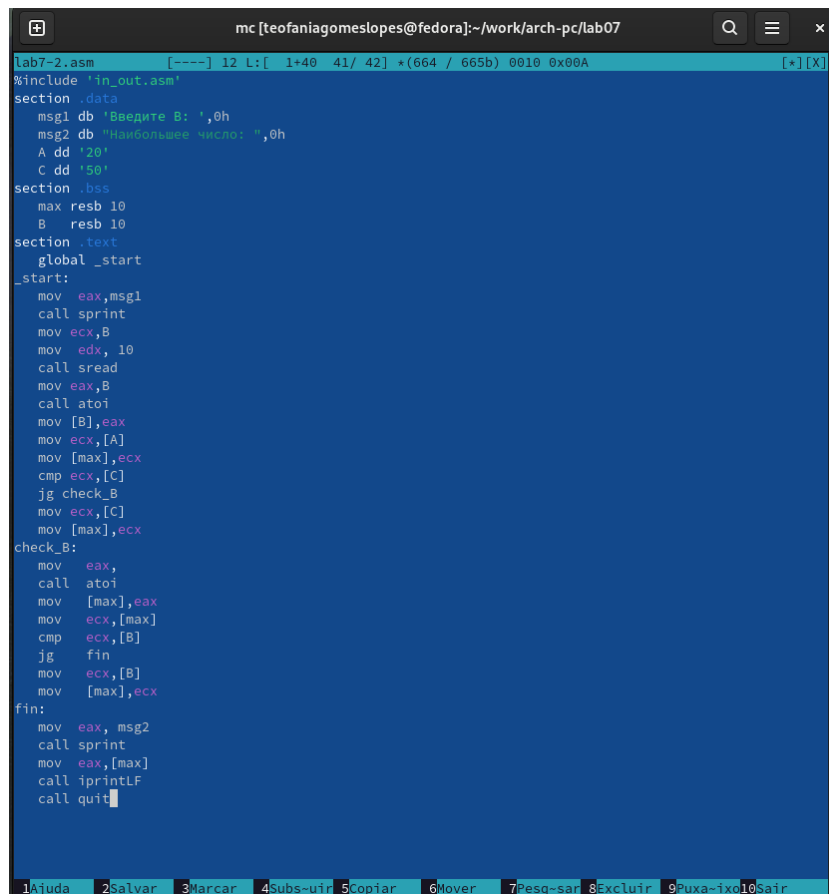
```

Рис. 3.12: Изучала файл

Строка 33: 0000001D-адрес в сегменте кода, B801000000-машинный код, `mov ebx,1`-присвоение переменной `ebx` значения 1.

Строка 34: 00000022-адрес в сегменте кода, B804000000-машинный код, `mov eax,4`-присвоение переменной `eax` значения 4.

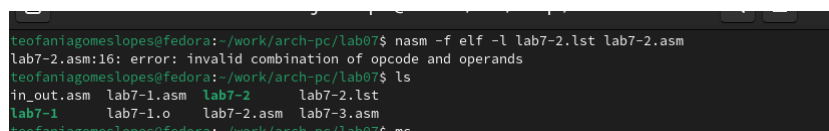
Открывала файл и удаляла один операндум (рис. 3.13).



```
lab7-2.asm [----] 12 L: [ 1+40 41/ 42] *(664 / 665b) 0010 0x00A [*][X]
#include "in_out.asm"
section .data
    msg1 db "Введите B: ",0h
    msg2 db "Наибольшее число: ",0h
    A dd '20'
    C dd '50'
section .bss
    max resb 10
    B resb 10
section .text
    global _start
_start:
    mov eax,msg1
    call sprint
    mov ecx,B
    mov edx,10
    call sread
    mov eax,B
    call atoi
    mov [B],eax
    mov ecx,[A]
    mov [max],ecx
    cmp ecx,[C]
    jg check_B
    mov ecx,[C]
    mov [max],ecx
check_B:
    mov eax,
    call atoi
    mov [max],eax
    mov ecx,[max]
    cmp ecx,[B]
    jg fin
    mov ecx,[B]
    mov [max],ecx
fin:
    mov eax,msg2
    call sprint
    mov eax,[max]
    call iprintLF
    call quit
```

Рис. 3.13: Удаляла операндум из файла

Транслировала с получением файла листинга (рис. 3.14).



```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf -l lab7-2.lst lab7-2.asm
lab7-2.asm:16: error: invalid combination of opcode and operands
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ls
in_out.asm lab7-1.asm lab7-2 lab7-2.lst
lab7-1 lab7-1.o lab7-2.asm lab7-3.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ mc
```

Рис. 3.14: Транслировала файл

При трансляции файла, выдается ошибка, но создаются исполнительный файл lab7-2 и lab7-2.lst

Снова открывала файл листинга и изучала его (рис. 3.15).

```

mc [teofaniagomeslopes@fedora]:~/work/arch-pc/lab07
lab7-2.lst [-----] 0 L:[181+21 202/219] *(12248/13190b) 0032 0x020 [*][X]
5 00000035 32300000 A dd '20'
6 00000039 35300000 C dd '50'
7
8 00000000 <res Ah> section .bss
9 0000000A <res Ah> max resb 10
10 B resb 10
11 section .text
12 global _start
13 000000E8 B8[00000000] _start:
14 000000ED E810FFFFFF mov eax,msg1
15 000000F2 B9[0A000000] call sprint
16 000000F8 mov ecx,B
17 000000FD E847FFFFFF mov edx
18 000000FC B8[0A000000] error: invalid combination of opcode and operands
19 00000101 E896FFFFFF call sread
20 00000106 A3[0A000000] mov eax,B
21 00000108 8B0D[35000000] call atoi
22 00000111 890D[00000000] mov [B],eax
23 00000117 3B0D[39000000] mov ecx,[A]
24 0000011D 7F0C cmp ecx,[C]
25 0000011F 8B0D[39000000] jg check_B
26 00000125 890D[00000000] mov ecx,[C]
27 check_B:
28 0000012B B8[00000000] mov eax,max
29 00000130 E867FFFFFF call atoi
30 00000135 A3[00000000] mov [max],eax
31 0000013A 8B0D[00000000] mov ecx,[max]
32 00000140 3B0D[0A000000] cmp ecx,[B]
33 00000146 7F0C jg fin
34 00000148 8B0D[0A000000] mov ecx,[B]
35 0000014E 890D[00000000] mov [max],ecx
36 fin:
37 00000154 B8[13000000] mov eax,msg2
38 00000159 E8B1FFFFFF call sprint
39 0000015E A1[00000000] mov eax,[max]
40 00000163 E81EFFFFFF call iprintLF
41 00000168 E86EFFFFFF call quit
1Ajuda 2Salvar 3Imprimir 4Substituir 5Copiar 6Mover 7Pesquisar 8Excluir 9Favoritos 10Sair

```

Рис. 3.15: Изучала файл с ошибкой

3.3 Задание для самостоятельной работы

ВАРИАНТ-12

1. Напишите программу нахождения наименьшей из 3 целочисленных переменных `a`, `b` и `c`. Значения переменных выбрать из табл. 7.5 в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу.

Создала новый файл (рис. 3.16).

```

teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ touch lab7-3.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$

```

Рис. 3.16: Создала файл командой touch

Открывала его и писала программу, которая выберет наименьшее число из трех (2 числа уже в программе, 3е вводится из консоли) (рис. 3.17).

```

mc [teofaniagomeslopes@fedora:~/work/arch-pc/lab07]
lab7-3.asm [-----] 0 L: [ 1+ 0 1/ 42] *(0 / 667b) 0037 0x025 [*][X]
#include "in_out.asm"
section .data
    msg1 db "Введите B: ",0h
    msg2 db "Наибольшее число: ",0h
    A dd '99'
    C dd '26'
section .bss
    max resb 10
    B resb 10
section .text
    global _start
_start:
    mov eax,msg1
    call sprint
    mov ecx,B
    mov edx,10
    call sread
    mov eax,B
    call atoi
    mov [B],eax
    mov ecx,[A]
    mov [max],ecx
    cmp ecx,[C]
    jg check_B
    mov ecx,[C]
    mov [max],ecx
check_B:
    mov eax,max
    call atoi
    mov [max],eax
    mov ecx,[max]
    cmp ecx,[B]
    jg fin
    mov ecx,[B]
    mov [max],ecx
fin:
    mov eax,msg2
    call sprint
1Ajuda 2Salvar 3Marcar 4Subs-uir 5Copiar 6Mover 7Pesq-sar 8Excluir 9Puxa-aixo10Sair

```

Рис. 3.17: Писала программу

Транслировала файл и смотрела на работу программы (рис. 3.18)

```

teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-3.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-3 lab7-3.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-3
Введите B: 29
Наименьшее число: 26
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$

```

Рис. 3.18: Смотрела на работу программы

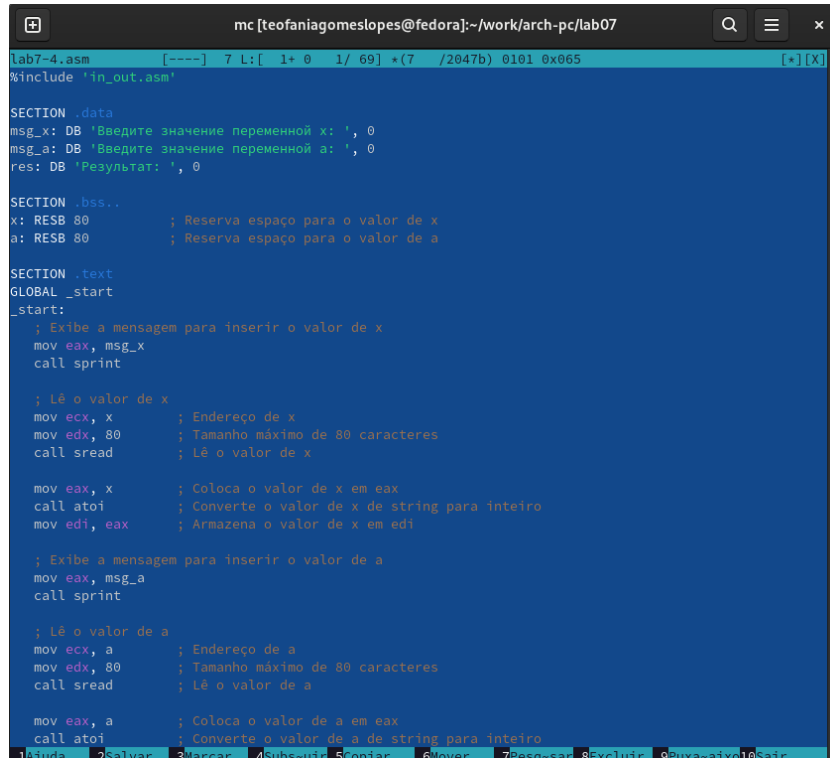
2. Напишите программу, которая для введенных с клавиатуры значений x и y вычисляет значение заданной функции $f(x, y)$ и выводит результат вычислений. Вид функции $f(x, y)$ выбрать из таблицы 7.6 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу для значений x и y из 7.6.

Создала новый файл (рис. 3.19)

```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ touch lab7-4.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$
```

Рис. 3.19: Создала файл командой touch

Открывала его и пишем программу, которая решит систему уравнений, при данных, введенных в консоль (рис. 3.20)



```
lab7-4.asm [----] 7 L: [ 1+ 0 1/ 69] *(7 /2047b) 0101 0x065 [*][X]
#include 'in_out.asm'

SECTION .data
msg_x: DB 'Введите значение переменной x: ', 0
msg_a: DB 'Введите значение переменной a: ', 0
res: DB 'Результат: ', 0

SECTION .bss..
x: RESB 80 ; Reserva espaço para o valor de x
a: RESB 80 ; Reserva espaço para o valor de a

SECTION .text
GLOBAL _start
_start:
; Exibe a mensagem para inserir o valor de x
mov eax, msg_x
call sprint

; Lê o valor de x
mov ecx, x ; Endereço de x
mov edx, 80 ; Tamanho máximo de 80 caracteres
call sread ; Lê o valor de x

mov eax, x ; Coloca o valor de x em eax
call atoi ; Converte o valor de x de string para inteiro
mov edi, eax ; Armazena o valor de x em edi

; Exibe a mensagem para inserir o valor de a
mov eax, msg_a
call sprint

; Lê o valor de a
mov ecx, a ; Endereço de a
mov edx, 80 ; Tamanho máximo de 80 caracteres
call sread ; Lê o valor de a

mov eax, a ; Coloca o valor de a em eax
call atoi ; Converte o valor de a de string para inteiro
```

Рис. 3.20: Писала программу

Транслировала файл и проверяла его работу (рис. 3.21)

```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-4
Введите значение переменной x: 3
Введите значение переменной a: 7
Результат: 21
```

Рис. 3.21: Проверяла работу программы

Транслировала файл и проверяем его работу (рис. 3.22)


```
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ ./lab7-4
Введите значение переменной x: 6
Введите значение переменной a: 4
Результат: 1
teofaniagomeslopes@fedora:~/work/arch-pc/lab07$ mc
```

Рис. 3.22: Проверка работы программы

4 Выводы

Я познакомилась с структурой файла листинга, изучили команды условного и безусловного перехода.

Список литературы