Отчёт по лабораторной работе %8

Дисциплина: ‘архитектура компьютеров’

Бабенко Роман Игоревич

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Изучить команды условного и безусловного переходов, приобрести навыки написания программ с использование переходов, а также познакомиться с назначением и структурой файла листинга.

2 Выполнение лабораторной работы

Создаём каталог lab08 и файл lab8-1.asm (рис. 2.1)

```
ribabenko@dk1n22 ~ $ mkdir ~/work/arch-pc/lab08
ribabenko@dk1n22 ~ $ cd ~/work/arch-pc/lab08
ribabenko@dk1n22 ~/work/arch-pc/lab08 $ touch lab8-1.asm
```

Рис. 2.1: Создание каталога и файла

Вводим в файл текст программы из листинга 8.1(рис. 2.2)

```
%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

 _label2:
  mov eax, msg2
  call sprintLF

 _label3:
  mov eax, msg3
  call sprintLF

 _end:
  call quit
```

Рис. 2.2: Перепечатывам текст программы

Создаём исполняемый файл и запускаем его (рис. 2.3)

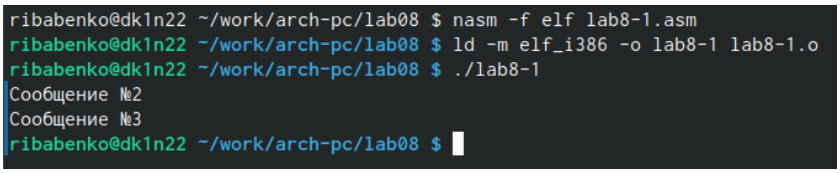

Рис. 2.3: Запуск программы

Изменим программу в соответствии с листингом 8.2 и проверяем его работу(рис. 2.4)

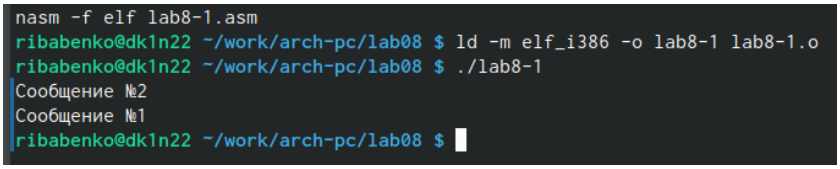

Рис. 2.4: Проверка изменённой программы

Снова изменяем текст программы так: (рис. 2.5)

```
%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 sprintLF
  jmp _end

 _label2:
  mov eax, msg2
  call sprintLF
  jmp _label1

 _label3:
  mov eax, msg3
  call sprintLF
 jmp _label2

 _end:
  call quit
```

Рис. 2.5: Изменяем программу

Проверяем нашу программу (рис. 2.6)

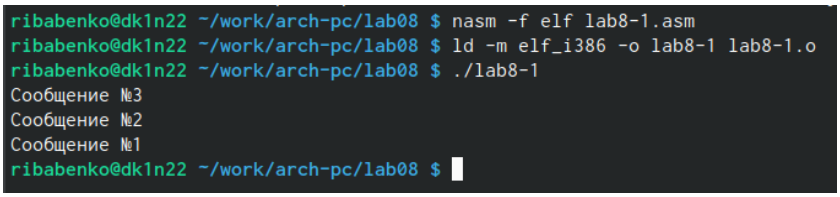

Рис. 2.6: Проверка вывода

Создаём файл lab8-2.asm (рис. 2.7)

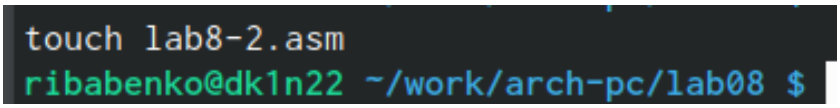

Рис. 2.7: Создание ещё одного файла

Переписываем в созданный файл программу из листинга 8.3 и провеярем его работу (рис. 2.8),(рис. 2.9)

Рис. 2.8: Проверка программы 8.2

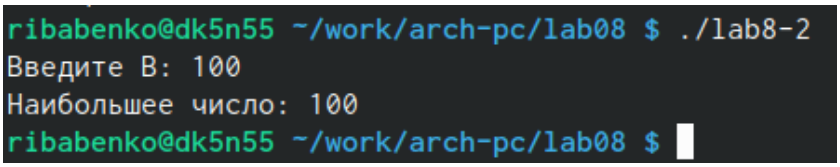

Рис. 2.9: Проверка программы 8.2

Создаём файл листинга и открываем его с помощью текстового редактора (рис. 2.10)

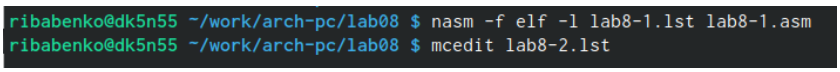

Рис. 2.10: Создание файла листинга

Изучаем формат и содержимое файла (рис. 2.11)

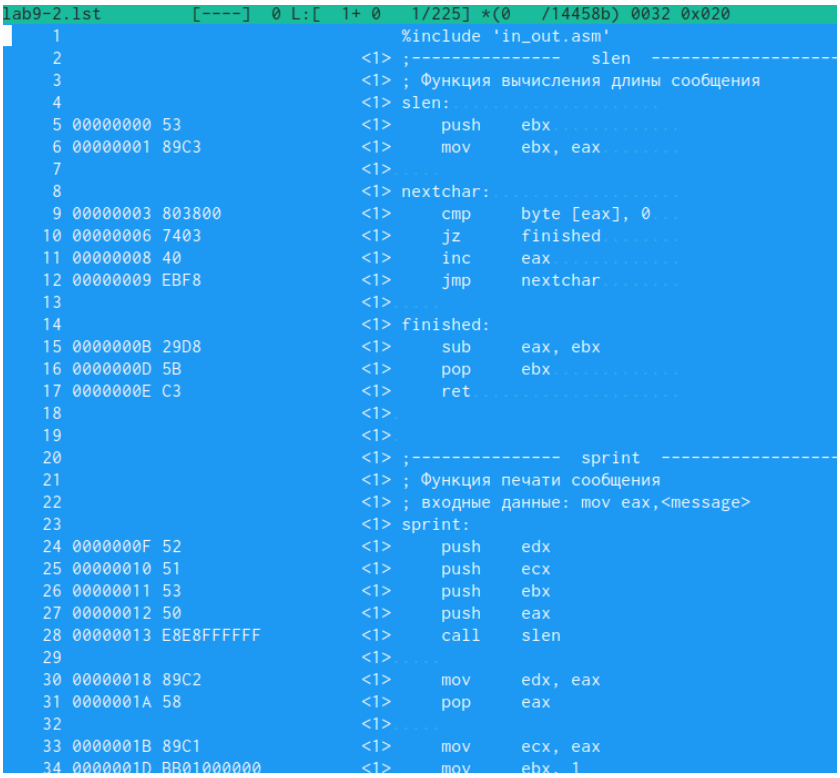

Рис. 2.11: Файл листинга

Открываем файл с программой lab8-2.asm и на строке 39 удаляем операнд.Выполняем трансляцию с получением файла листинга (рис. 2.12)

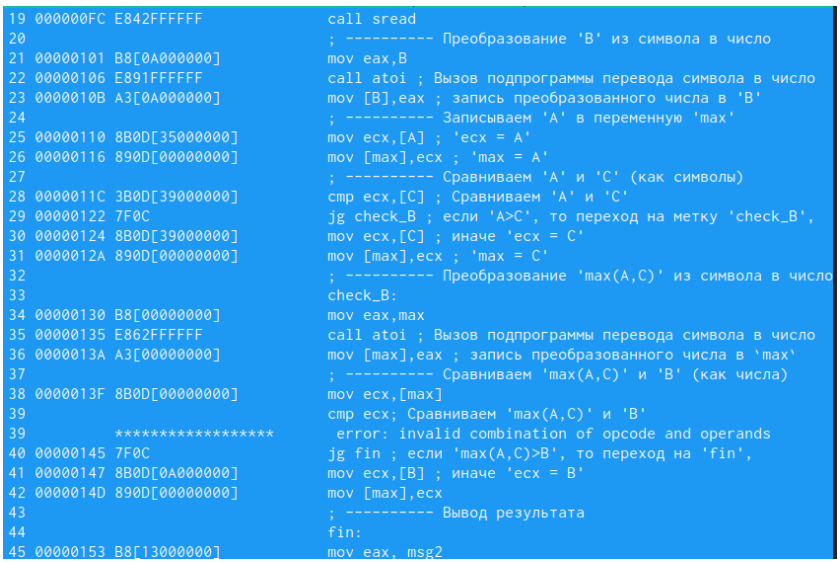

Рис. 2.12: Трансляция файла листинга после удаления операнда

Открываем текстовый редактор mcedit. (рис. 2.13)

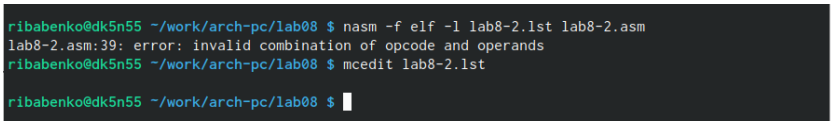

Рис. 2.13: Текстовый редактор mcedit

#Задания для самостоятельной работы

Создаём файл lab8-3.asm и напишем программу для нахождения наименьшей из трёх переменных (рис. 2.14)

```
%include 'in_out.asm'
section .data
msg2 db "Наименьшее  число: ",0h
A dd '45'
B dd '67'
C dd '15'
section .bss
min resb 10
section .text
global _start

_start:
mov ecx,[A]
mov [min],ecx
cmp ecx,[C]
jl check_B
mov ecx,[C]
mov [min],ecx

check_B:
mov eax,min
call atoi
mov [min],eax
mov ecx,[min]
cmp ecx,[B]
jl fin
mov ecx,[B]
mov [min],ecx

fin:
mov eax, msg2
call sprint
mov eax,[min]
call iprintLF
call quit
```

Рис. 2.14: Программа

Проверка написанной программы (рис. 2.15)

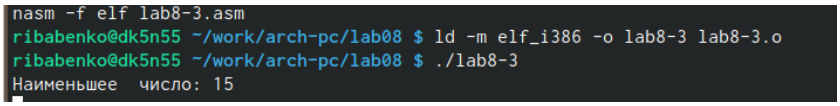

Рис. 2.15: Результат программы

Напишем программу для вычисления значения функции по 7 варианту (рис. [-fig. 2.16) и (рис. 2.17)

```
%include 'in_out.asm'
section .data
vv1 db "Введите x: ",0h
vv2 db "Введите a: ",0h
clishe db "Результат: ",0h

section .bss
x resb 10
a resb 10

section .text
global _start

_start:

mov eax,vv1
call sprint

mov ecx,x
mov edx,10
call sread

mov eax,x
call atoi
mov [x],eax

mov eax,vv2
call sprint

mov ecx,a
mov edx,10
call sread

mov eax,a
call atoi
mov [a],eax

mov ecx,[x]
cmp ecx,[a]
je check_N
```

Рис. 2.16: Написанная програма(1)

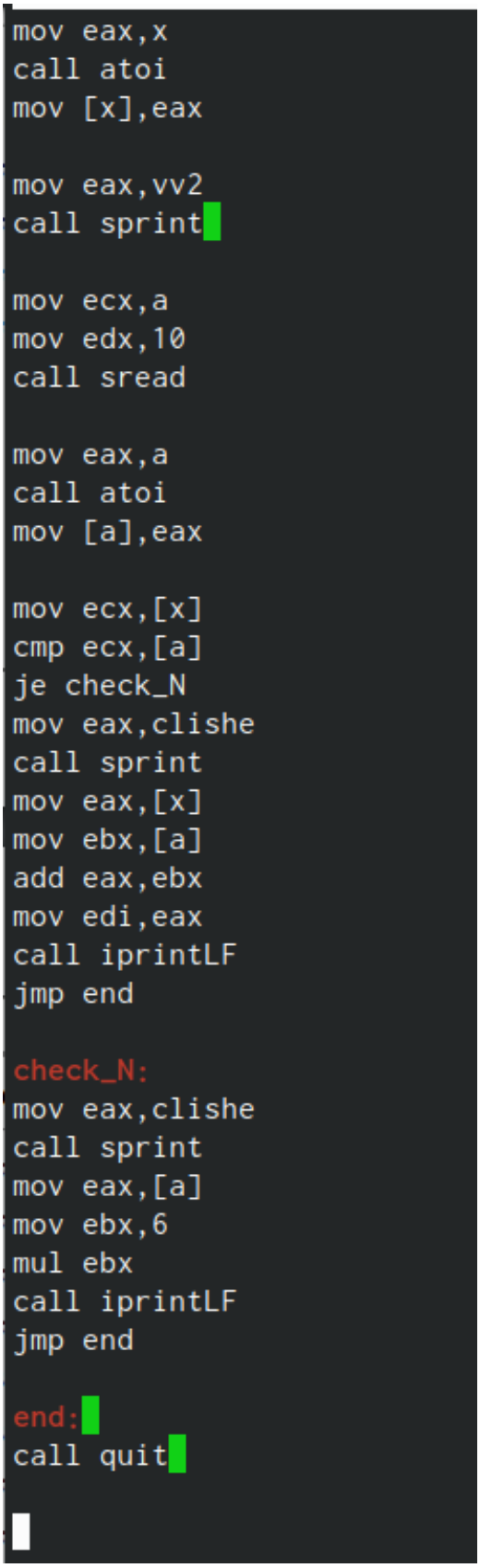

Рис. 2.17: Написанная програма(2)

Проверяем результат выполнения написанной программы (рис. 2.18)

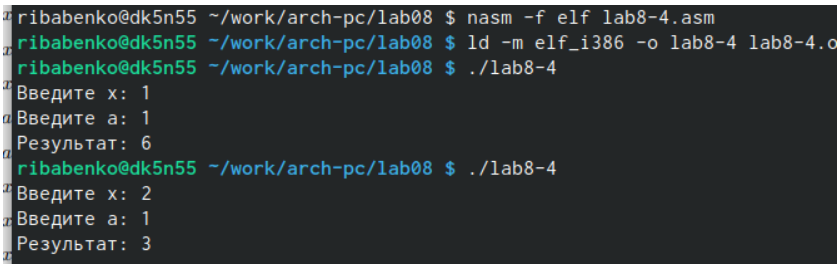

Рис. 2.18: Проверка программы

3 Выводы

В ходе выполнения лабораторной раборы я научился пользоваться командами условного и безусловного переходов, приобрёл навыки написания программ с использование переходов, а также познакомился с назначением и структурой файла листинга.