Отчет по лабораторной работе №7

Дисциплина: архитектура компьютера

Козина Дарья Александровна

Содержание

[Цель работы 4](#__RefHeading___Toc545_776834727)

[Задание 5](#__RefHeading___Toc547_776834727)

[Выполнение лабораторной работы 6](#__RefHeading___Toc549_776834727)

[Реализация переходов в NASM 6](#__RefHeading___Toc551_776834727)

[Изучение структуры файла листинга 9](#__RefHeading___Toc553_776834727)

[Выполнение заданий для самостоятельной работы 11](#__RefHeading___Toc555_776834727)

[Вывод 17](#__RefHeading___Toc557_776834727)

# Цель работы

Изучить команды условного и безусловного переходов. Приобрести навыки написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

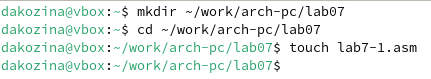
# Задание

1. Реализовать переходы в NASM;
2. Изучить структуры файла листинга;
3. Выполнение заданий для самостоятельной работы.

# Выполнение лабораторной работы

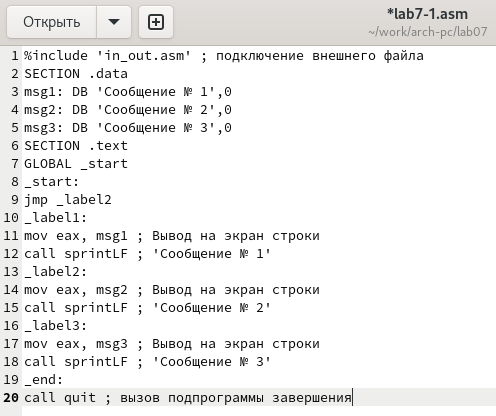
## Реализация переходов в NASM

Создадим каталог для программ лабораторной работы, перейдем в созданный каталог и создадим файл lab7-1.asm (рис. [-@fig:001]).



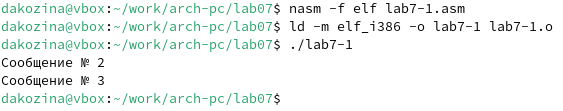
Создание каталога и файла

С помощью редактора gedit введем в файл lab7-1.asm текст программы в соответствии с листингом 7.1 (рис. [-@fig:002]).



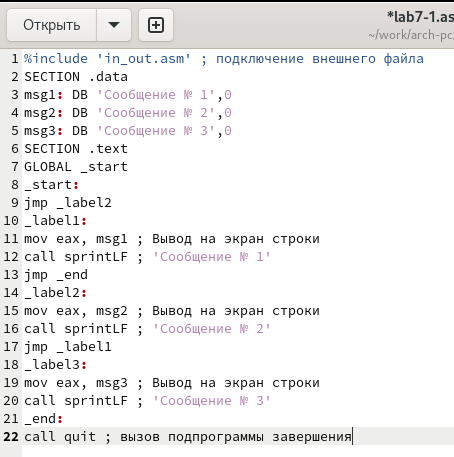
Редактирование файла

Создадим исполняемый файл и запустим его (рис. [-@fig:003]).



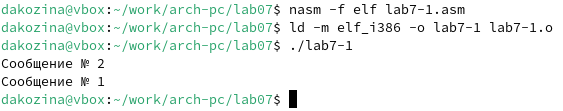
Создание и запуск файла

С помощью редактора gedit изменим текст программы так, чтобы сначала выводилось ‘Сообщение № 2’, потом ‘Сообщение № 1’ в соответствии с листингом 7.2 (рис. [-@fig:004]).



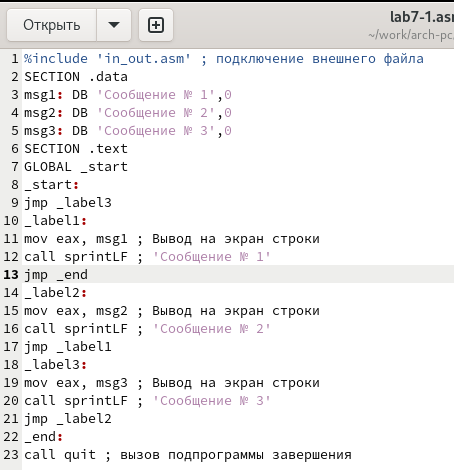
Редактирование файла

Создадим исполняемый файл и запустим его (рис. [-@fig:005]).



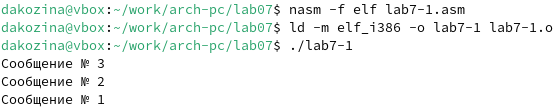
Создание и запуск файла

С помощью редактора gedit изменим текст программы, чтобы выводилось сначала ‘Сообщение № 3’, потом ‘Сообщение № 2’, а в конце ‘Сообщение № 1’ (рис. [-@fig:006]).



Редактирование файла

Создадим исполняемый файл и запустим его (рис. [-@fig:007]).



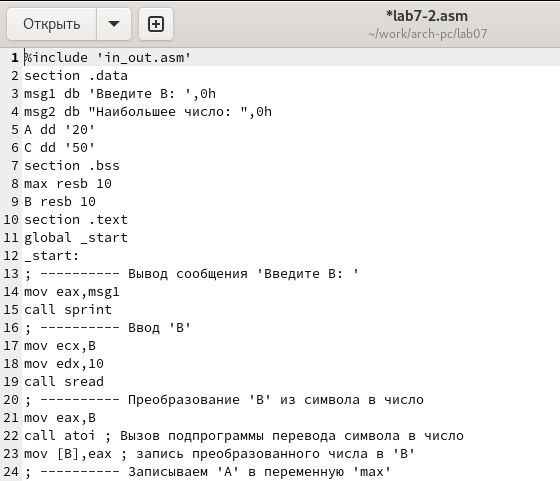
Создание и запуск файла

Создадим файл lab7-2.asm в каталоге, созданном в начале лабораторной работы (рис. [-@fig:008]).

Создание файла

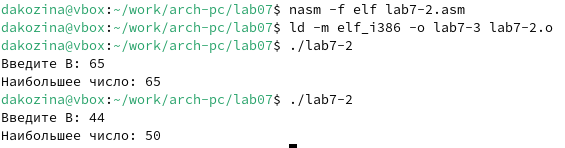
Создание файла

С помощью редактора gedit изменим текст программы в соответствии с листингом 7.3 (рис. [-@fig:009]).



Редактирование файла

Создадим исполняемый файл и запустим его(рис. [-@fig:010]).



Создание и запуск файла

## Изучение структуры файла листинга

С помощью команды nasm и ключа -l создадим файл листинга для программы из файла lab7-2.asm (рис. [-@fig:011]).

Создание файла листинга

Создание файла листинга

Откроем файл листинга lab7-2.lst в редакторе mcedit. Внимательно ознакомимся с его форматом и содержанием, объясним содержимое трех строк файла.

1. Эта строка находится на 21 месте, ее адрес “00000101”, Машинный код - В8 [0A000000], а mov eax,B - исходный текст программы, означающий что в регистр eax мы вносим значения переменной B (рис. [-@fig:012]).

Строка 21

Строка 21

1. Эта строка находится на 35 месте, ее адрес “00000130”, Машинный код - E867FFFFFF, а call atoi - исходный текст программы, означающий что символ лежащий в строке выше переводится в число (рис. [-@fig:013]).

Строка 35

Строка 35

1. Эта строка находится на 47 месте, ее адрес “00000163”, Машинный код - A1[00000000], а mov eax,[max] - исходный текст программы, означающий что число хранившееся в переменной max записывается в регистр eax (рис. [-@fig:014]).

Строка 47

Строка 47

С помощью редактора gedit откроем файл с программой lab7-2.asm и в строке ‘mov eax, max’ удалим ‘max’ (рис. [-@fig:015]).

Удаление операнда

Удаление операнда

Выполним трансляцию с получением файла листинга (рис. [-@fig:016]).

Трансляция файла

Трансляция файла

В файле листинга показывает где именно ошибка и с чем она связана (рис. [-@fig:017]).

Ошибка в файле листинга

Ошибка в файле листинга

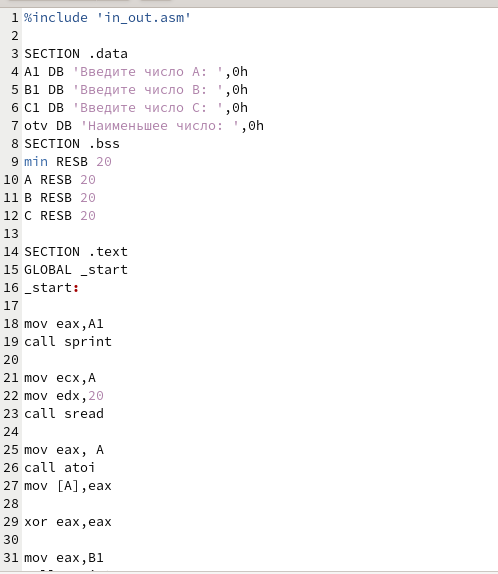
# Выполнение заданий для самостоятельной работы

1. Создадим файл lab7-3.asm (рис. [-@fig:017]).

Создание файла

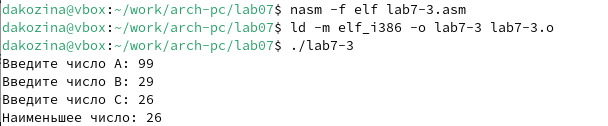
Создание файла

В редакторе gedit напишем программу нахождения наименьшей из 3 целочисленных переменных (рис. [-@fig:017]).



Редактирование файла

Создадим исполняемый файл и запустим его (рис. [-@fig:017]). Мой вариант 12, я ввожу значения a, b, c в соответствии с этим вариантом.



Создание и запуск файла

**Листинг программы**

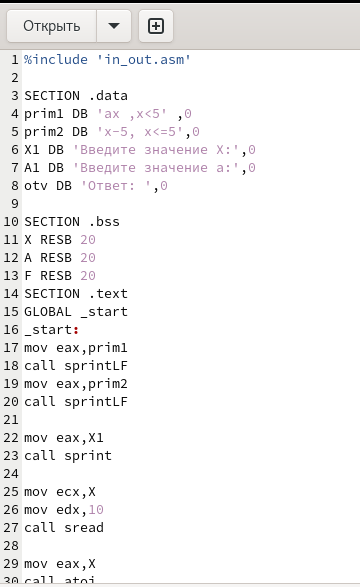
%include 'in\_out.asm'  
  
SECTION .data  
A1 DB 'Введите число А: ',0h  
B1 DB 'Введите число B: ',0h  
C1 DB 'Введите число С: ',0h  
otv DB 'Наименьшее число: ',0h  
SECTION .bss  
min RESB 20  
A RESB 20  
B RESB 20  
C RESB 20  
  
SECTION .text  
GLOBAL \_start  
\_start:  
  
mov eax,A1  
call sprint  
  
mov ecx,A  
mov edx,20  
call sread  
  
mov eax, A  
call atoi  
mov [A],eax  
  
xor eax,eax  
  
mov eax,B1  
call sprint  
  
mov ecx,B  
mov edx,20  
call sread  
  
mov eax,B  
call atoi  
mov [B],eax  
  
xor eax,eax  
  
mov ecx, [A]  
mov [min],ecx  
mov ecx,[min]  
  
cmp ecx,[B]  
jl check\_C  
mov ecx, [B]  
mov [min],ecx  
  
check\_C:  
  
mov eax,C1  
call sprint  
  
mov ecx,C  
mov edx,10  
call sread  
  
mov eax,C  
call atoi  
mov [C],eax  
  
xor eax,eax  
  
  
mov ecx,[min]  
cmp ecx,[C]  
jl fin  
mov ecx,[C]  
mov [min],ecx  
  
  
fin:  
mov eax, otv  
call sprint  
mov eax, [min]  
call iprintLF  
call quit

1. Создадим файл lab7-4.asm (рис. [-@fig:017]).

Создание файла

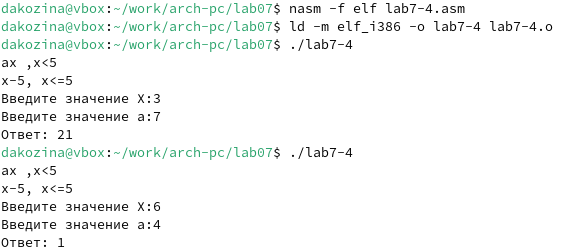
Создание файла

В редакторе gedit напишем программу для решения функции варианта 12 (рис. [-@fig:017]).



Редактирование файла

Создадим исполняемый файл и запустим его (рис. [-@fig:017]).



Создание и запуск файла

**Листинг программы**

%include 'in\_out.asm'  
  
SECTION .data  
prim1 DB 'ax ,x<5' ,0  
prim2 DB 'x-5, x<=5',0  
X1 DB 'Введите значение Х:',0  
A1 DB 'Введите значение a:',0  
otv DB 'Ответ: ',0  
  
SECTION .bss  
X RESB 20  
A RESB 20  
F RESB 20  
SECTION .text  
GLOBAL \_start  
\_start:  
mov eax,prim1  
call sprintLF  
mov eax,prim2  
call sprintLF  
  
mov eax,X1  
call sprint  
  
mov ecx,X  
mov edx,10  
call sread  
  
mov eax,X  
call atoi  
mov [X],eax  
  
mov eax,A1  
call sprint  
  
mov ecx,A  
mov edx,10  
call sread  
  
mov eax,A  
call atoi  
mov [A],eax  
  
mov ecx,[X]  
mov [F],ecx  
mov eax, [X]  
cmp ecx,5  
jl check\_or  
mov edx,5  
sub ecx,edx  
mov [F],ecx  
jmp fin  
  
check\_or:  
mov eax,[A]  
mov ebx,[X]  
mul ebx  
mov [F],eax  
  
fin:  
mov eax,otv  
call sprint  
mov eax,[F]  
call iprintLF  
call quit

# Вывод

В ходе лабораторной работы мы изучили команды условного и безусловного переходов. Приобрели навыки написания программ с использованием переходов. Познакомились с назначением и структурой файла листинга.