Лабораторная работа №7

Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Жукова Арина Александровна

Содержание

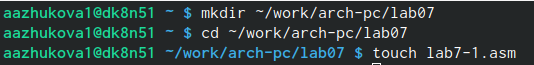
# 1 Цель работы

Цель данной лабораторной работы изучить команды условного и безусловного переходов, приобрести навыки написания программ с использованием переходов, познакомство с назначением и структурой файла листинга.

# 2 Выполнение лабораторной работы

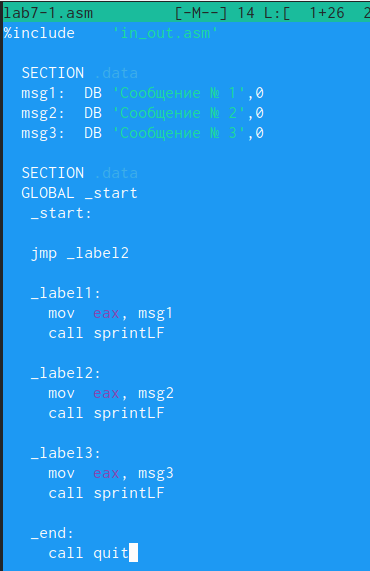
## 2.1 Реализация переходов в NASM

1. Создадим файлы lab7-1.asm в каталоге для программ лабораторной работы №7 (рис. ??).



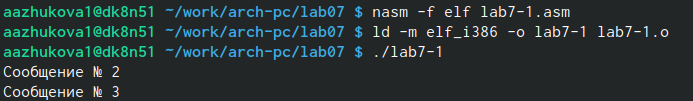
Создание файла

1. Введём в файл lab7-1.asm текст программы из листинга 7.1 (рис. ??).



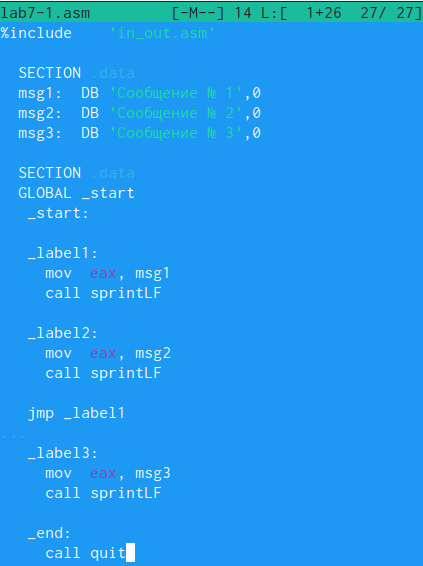
Ввод текста программы

Создаём исполняемый файл и проверяем его работу (рис. ??).



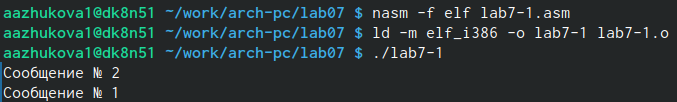
Создание и проверка работы исполняемого файла

Изменение текста программы файла так, чтобы программа сначала ‘Сообщение № 2’, потом ‘Сообщение № 1’ и завершала работу (рис. ??).



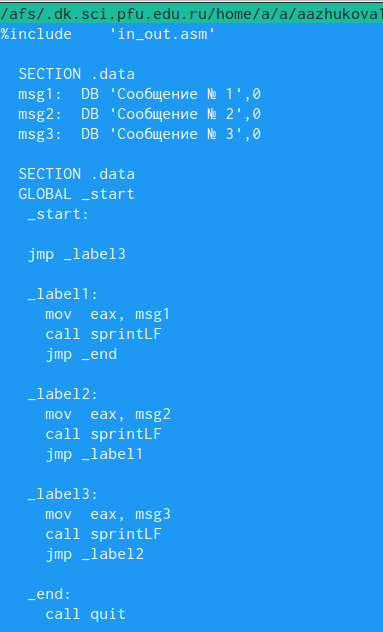
Редактирование текста программы

Создание исполняемого файла и проверка его работы (рис. ??).



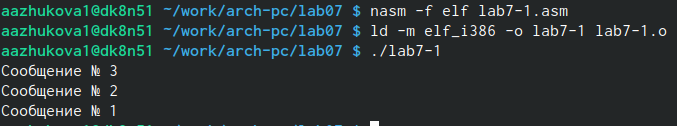
Проверка работы программы

Изменим текст программы так, чтобы программа сначала ‘Сообщение № 3’, ‘Сообщение № 2’, потом ‘Сообщение № 1’ и завершала работу (рис. ??).



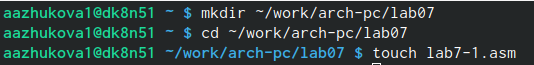
Редактирование текста программы

Создание исполняемого файла и проверка его работы (рис. ??).



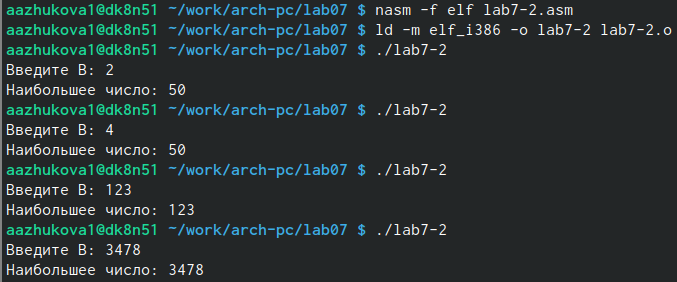
Проверка работы программы

1. Создадим файл lab7-2.asm в каталоге ~/work/arch-pc/lab07. Введём текст программы из листинга 7.3 в lab7-2.asm (рис. ??).



Создание файла

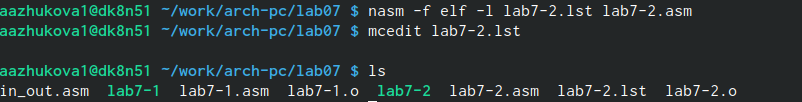
Проверим работу программы, введя различные значения B (рис. ??).



Проверка работы программы

## 2.2 Изучение структуры файлы листинга

1. Создадим файл листинга для программы из файла lab7-2.asm, указав ключ -l и задав имя файла листинга в командной строке (рис. ??).



Создание файла листинга

Опишем приведенные на рисунке 2.11 строки (рис. ??).



Строки файла листинга

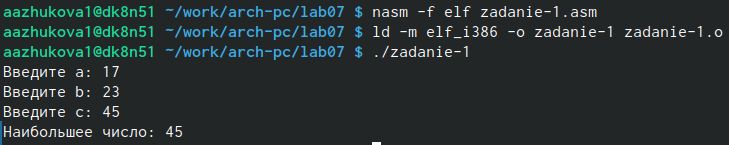
1. 32 - номер строки файла листинга, 0000001В - адрес (смещение машинного кода от начала текущего сегмента), 89C1 - машинный код, представляющая собой ассемблированную исходную строку в виде шестнадцатеричной последовательности, mov ecx, eax - исходный тест программы (приравнивающий значение ecx к eax).
2. 33 - номер строки файла листинга, 0000001D - адрес (смещение машинного кода от начала текущего сегмента), 8BB01000000 - машинный код, представляющая собой ассемблированную исходную строку в виде шестнадцатеричной последовательности, mov ebx, 1 - исходный тест программы (приравнивающий значение ebx к 1).
3. 34 - номер строки файла листинга, 00000022- адрес (смещение машинного кода от начала текущего сегмента), 8B804000000 - машинный код, представляющая собой ассемблированную исходную строку в виде шестнадцатеричной последовательности, mov ebx, 1 - исходный тест программы (приравнивающий значение eax к 4).

## 2.3 Задания для самостоятельной работы

1. Программа нахождения наименьшей из 3 целочисленных переменных a, b и c.

%include 'in\_out.asm'  
section .data  
 msg db "Наибольшее число: ",0h  
 msgA db "Введите a: ",0h  
 msgB db "Введите b: ",0h  
 msgC db "Введите c: ",0h  
section .bss  
 max resb 10  
 A resb 10  
 B resb 10  
 C resb 10  
section .text  
 global \_start  
\_start:  
; ------- Ввод A, B, C --------  
 mov eax,msgA  
 call sprint  
   
 mov eax,A  
 mov edx,10  
 call sread  
   
 mov eax,msgB  
 call sprint  
   
 mov ebx,B  
 mov edx,10  
 call sread  
   
 mov eax,msgC  
 call sprint  
   
 mov ecx,C  
 mov edx,10  
 call sread  
; -------- Преобразование ---------  
 mov eax,B  
 call atoi  
 mov [B],eax  
; -------- Записываем А в перенную max --------  
 mov ecx,[A]  
 mov [max],ecx ; max = A  
; ------- Сравниваем A и C -------------  
 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, msg  
 call sprint  
 mov eax,[max]  
 call iprintLF  
 call quit

Результат (рис. ??).

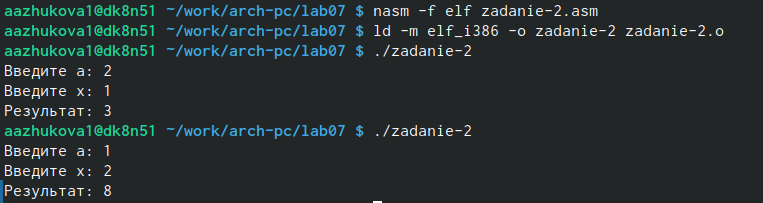


Результат работы программы

1. Программа, которая для введенных с клавиатуры значений x и a вычисляет значение заданной функции f(x) и выводит результат вычислений (2a − x, x < a, 8, x ≥ a).

%include 'in\_out.asm'  
section .data  
 result\_msg db "Результат: ",0h  
 msgA db "Введите a: ",0h  
 msgX db "Введите x: ",0h  
section .bss  
 a resb 10  
 x resb 10  
 result resd 1  
section .text  
 global \_start  
\_start:  
; ------- Ввод A --------  
 mov eax,msgA  
 call sprint  
   
 mov ecx,a  
 mov edx,10  
 call sread  
   
; ------- Преобразование A ------  
 mov eax,a  
 call atoi   
 mov [a],eax  
   
; ------- Ввод X ------------  
 mov eax,msgX  
 call sprint  
   
 mov ecx,x  
 mov edx,10  
 call sread  
   
; -------- Преобразование X ---------  
 mov eax,x  
 call atoi  
 mov [x],eax  
   
; --------- Сравнение ---------  
 mov eax, [a]  
 cmp eax, [x]  
 jge less\_A  
   
 mov eax, result\_msg  
 call sprint  
 mov eax,8  
 call iprintLF  
 call quit  
   
less\_A:  
 mov eax,[a]  
 mov ebx,2  
 mul ebx  
 sub eax, [x]  
 mov [result], eax  
   
 mov eax, result\_msg  
 call sprint  
 mov eax, [result]  
 call iprintLF  
   
 call quit

Результат (рис. ??).



Результат работы программы

# 3 Выводы

Нами были изучены команды условного и безусловного переходов, приобретены навыки написания программ с использованием переходов, мы познакомились с назначением и структурой файла листинга.

# Список литературы