Отчёт по лабораторной работе №7

Дисциплина:архитектура компьютера

Жернаков Данила Иванович

Содержание

# 1 Цель работы

Изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

# 2 Задание

1. Реализация переходов в NASM
2. Изучение структуры файла листинга
3. Задание для самостоятельной работы

# 3 Теоретическое введение

Для реализации ветвлений в ассемблере используются так называемые команды передачи управления или команды перехода. Можно выделить 2 типа переходов:

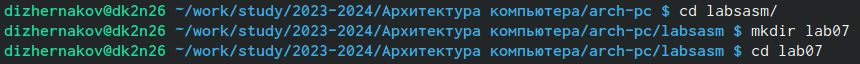
• Условный переход – выполнение или не выполнение перехода в определенную точку программы в зависимости от проверки условия.

• Безусловный переход – выполнение передачи управления в определенную точку про- граммы без каких-либо условий.

# 4 Выполнение лабораторной работы

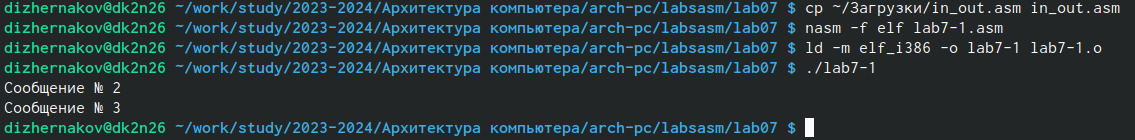
## 4.1 Реализация переходов в NASM

Создаю каталог для программ для лабораторной работе №7, перехожу в него и создаю файл lab7-1.asm (рис. ??).



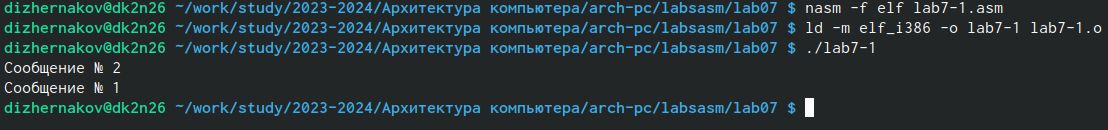
Создание каталога и файла в ней

Ввожу в файл lab7-1.asm текст программы с использованием функции jmp (рис. ??). Создаю исполняемый файл и запускаю его (рис. ??).



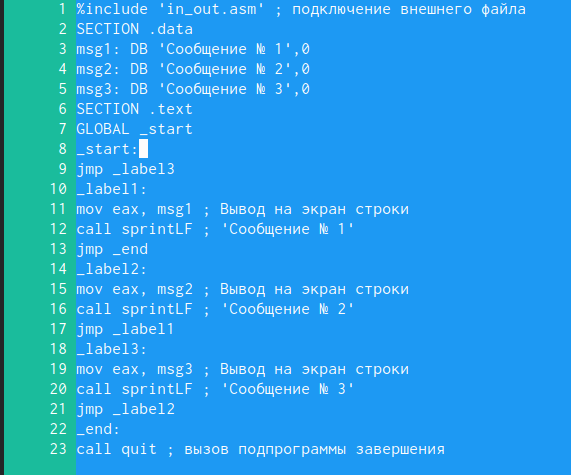
Исполнение программы

Изменяю программу таким образом, чтобы она выводила сначала ‘Сообщение No 2’, потом ‘Сообщение No 1’ и завершала работу (рис. ??).



Исполнение программы

Изменяю программу таким образом, чтобы она выводила сначала ‘Сообщение No 3’, потом ‘Сообщение No 2’, потом ‘Сообщение No 1’ и завершала работу (рис. ??).



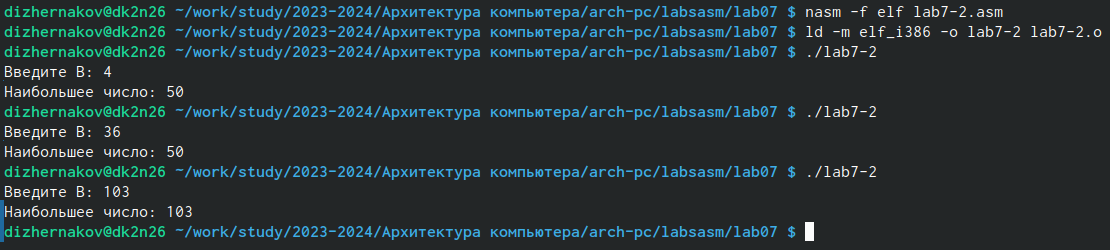
Редактирование файла

Создаю исполняемый файл и проверяю корректность работы программы (рис. ??). Программа отработала корректно.



Исполнение программы

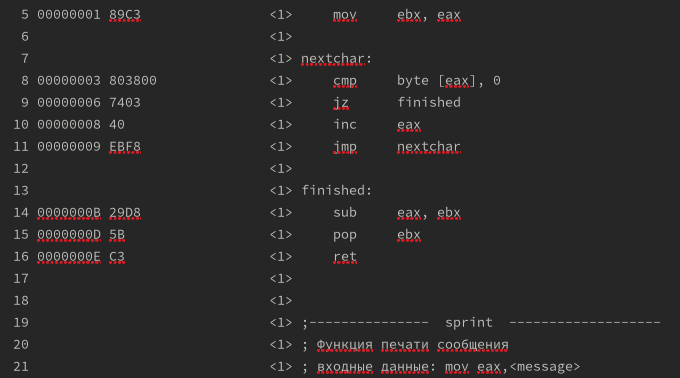
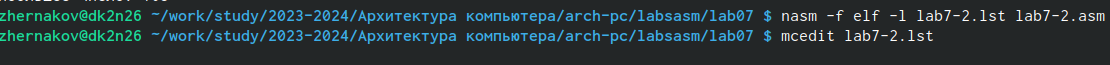
Создаю файл lab7-2.asm Ввожу в созданны файл текст программы, которая определяет и выводит на экран наибольшую из 3 целочисленных переменных:A, B и C Создаю исполняемый файл и проверяю его работу для разных значений B (рис. ??). Программа сработала корректно.



Исполнение программы для разных значений B

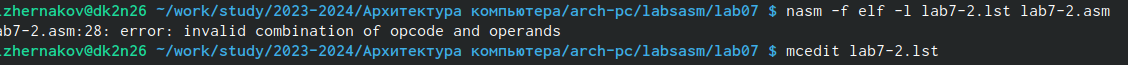
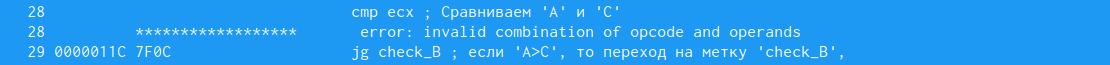
## 4.2 Изучение структуры файла листинга

Создание файла листинга и его просмотр в текстовом редакторе(рис. ??).

1. В строке 5 содержится собственно номер строки[5], адрес[00000001], машинный код[89C3] и содержимое строки кода[mov ebx, eax].
2. В строке 11 содержится собственно номер строки[11], адрес[00000009], машинный код[EBF8] и содержимое строки кода[jmp nextchar].
3. В строке 14 содержится собственно номер строки[14], адрес[0000000B], машинный код[29D8] и содержимое строки кода[sub eax, ebx].

Открываю файл lab7-2.asm и удаляю в инструкции cmp вторгй операнд (рис. ??). Открытие файла листинга после трансляции (рис. ??). Если в коде появляется ошибка, то её описание появится в файле листинга.

## 4.3 Выполнение заданий для самостоятельной работы (вар. 16)

Создаю файл lab7-3.asm, пишу в нём программу для нахождения наименьшей из трёх целочисленных переменных a, b и c.

Текст программы в файле lab7-3.asm:

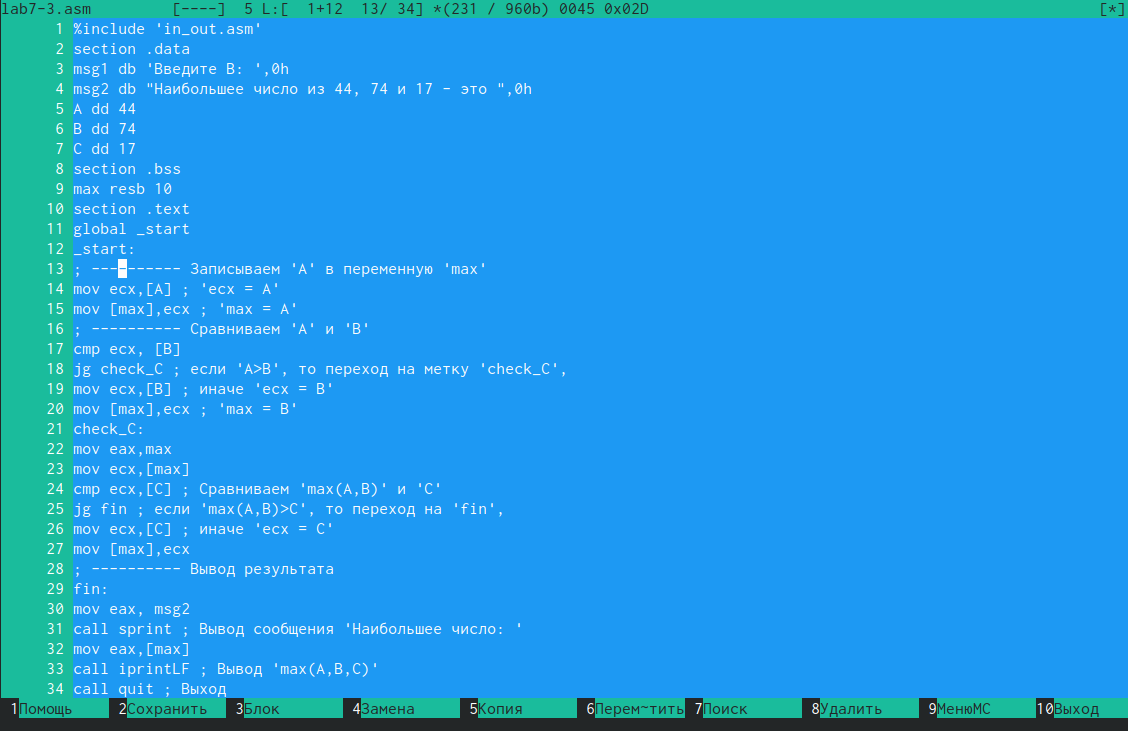
%include 'in\_out.asm'  
section .data  
msg1 db 'Введите B: ',0h  
msg2 db "Наибольшее число из 44, 74 и 17 - это ",0h  
A dd 44  
B dd 74  
C dd 17  
section .bss  
max resb 10  
section .text  
global \_start  
\_start:  
; ---------- Записываем 'A' в переменную 'max'  
mov ecx,[A] ; 'ecx = A'  
mov [max],ecx ; 'max = A'  
; ---------- Сравниваем 'A' и 'B'  
cmp ecx, [B]  
jg check\_C ; если 'A>B', то переход на метку 'check\_C',  
mov ecx,[B] ; иначе 'ecx = B'  
mov [max],ecx ; 'max = B'  
check\_C:  
mov eax,max  
mov ecx,[max]  
cmp ecx,[C] ; Сравниваем 'max(A,B)' и 'C'  
jg fin ; если 'max(A,B)>C', то переход на 'fin',  
mov ecx,[C] ; иначе 'ecx = C'  
mov [max],ecx  
; ---------- Вывод результата  
fin:  
mov eax, msg2  
call sprint ; Вывод сообщения 'Наибольшее число: '  
mov eax,[max]  
call iprintLF ; Вывод 'max(A,B,C)'  
call quit ; Выход

Создаю исполняемый файл и проверяю его работу (рис. ??). Программа отработала корректно.



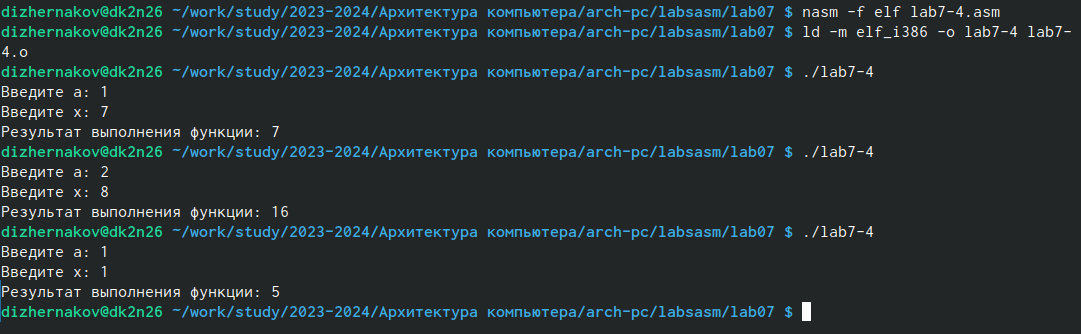
Исполнение программы

Создаю файл lab7-4.asm, пишу в нём програму, которая для введённых с клавиатуры значений x и a вычисляет значение функции f(x), которая равна xa при x>=4, когда x < 4, то х+4 и выводит результат вычислений.



Написание программы

Создаю исполняемый файл и проверяю его работу для пар x и a (1,1) и (7,1) (рис. ??). Программа отработала верно.



Исполнение программы

# 5 Выводы

В ходе выполнения лабораторной работы я освоил принципы условного и безусловного перехода в NASM.