Отчёт по лабораторной работе №7

Дисциплина: архитектура компьютера

Назармамадов Умед Джамшедович

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 файлы листинга.
3. Задание для самостоятельной работы.

# 3 Теоретическое введение

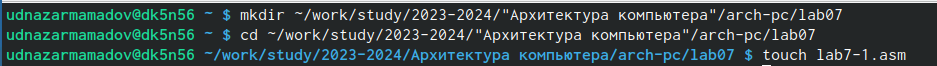
Для реализации ветвлений в ассемблере используются так называемые команды передачи управления или команды перехода. Можно выделить 2 типа переходов:

1. Условный переход - выполнение или не выполнение перехода в определенную точку программы в зависимости от проверки условия.
2. Безусловный переход - выполнение передачи управления в определенную точку программы без каких-либо условий.

Безусловный переход выполняется инструкцией jmp. Инструкция cmp является одной из инструкций, которая позволяет сравнить операнды и выставляет флаги в зависимости от результата сравнения. Инструкция cmp является командой сравнения двух операндов и имеет такой же формат, как и команда вычитания. Листинг (в рамках понятийного аппарата NASM) — это один из выходных файлов, создаваемых транслятором. Он имеет текстовый вид и нужен при отладке программы, так как кроме строк самой программы он содержит дополнительную информацию.

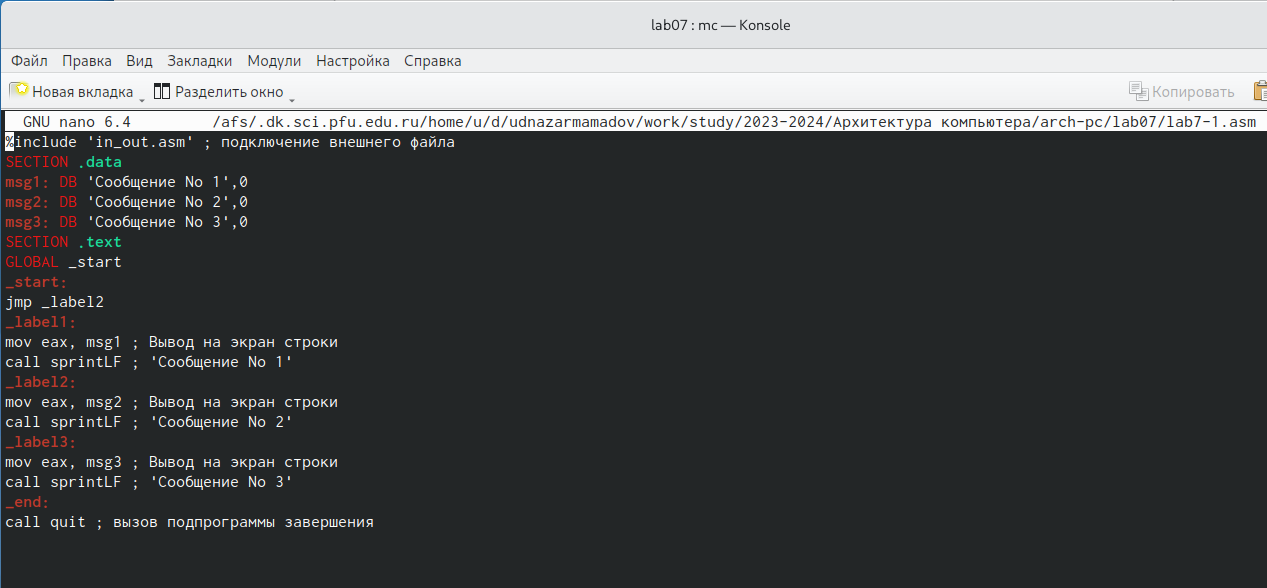
# 4 Выполнение лабораторной работы

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



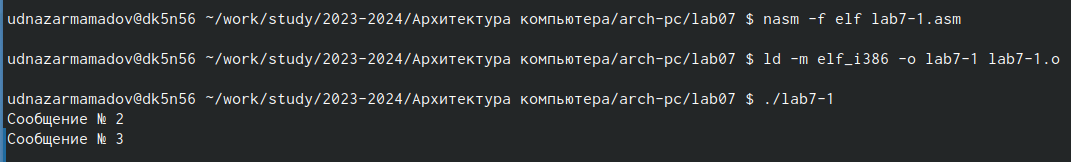
Создание файла

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



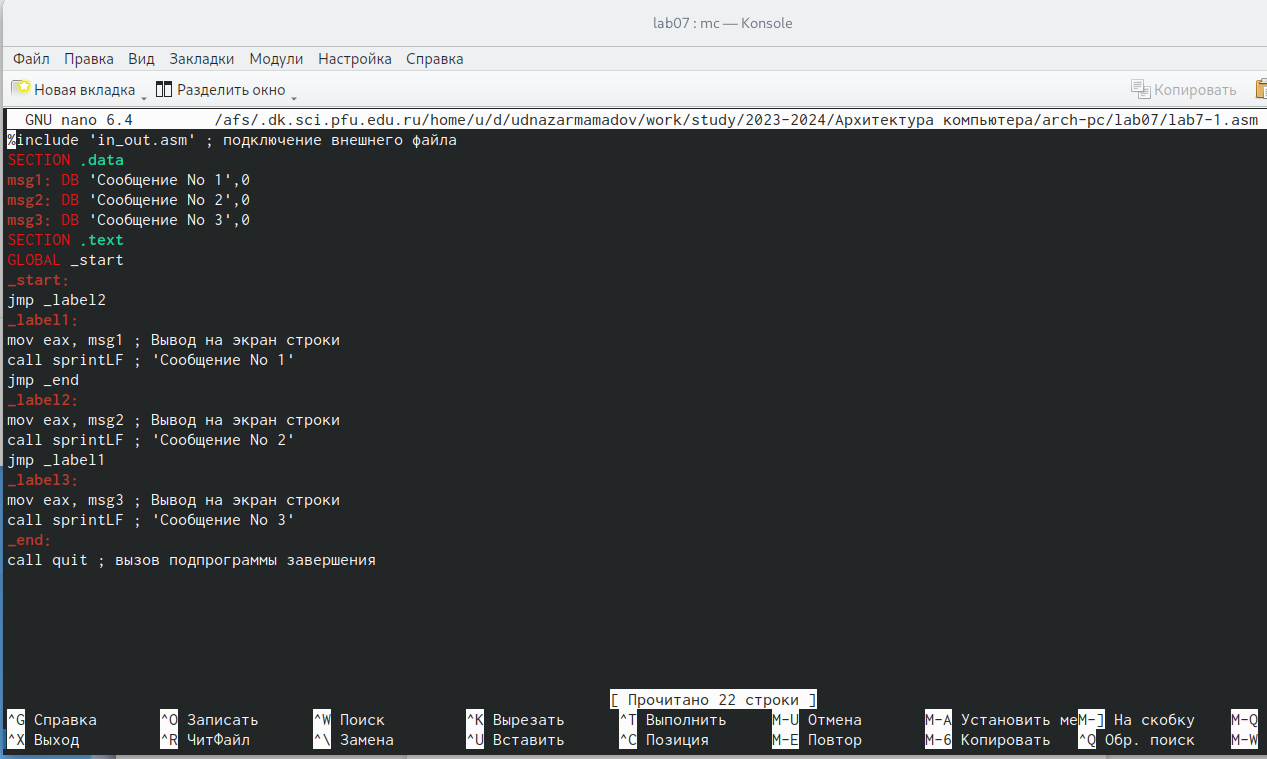
Редактирование файла

Создаю исполняемый файл и запускаю его (рис. -??).



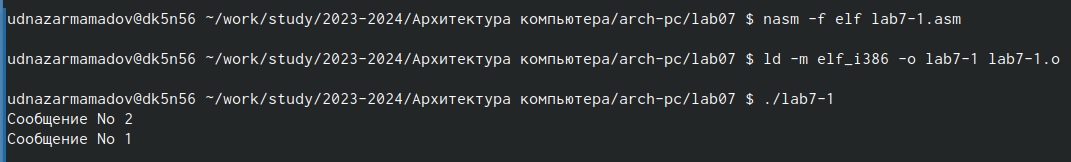
Запуск файла

Изменю программу таким образом, чтобы она выводила сначала ‘Сообщение № 2’, потом ‘Сообщение № 1’ и завершала работу. Для этого изменяю текст программы в соответствии с листингом 7.2 (рис. -??).



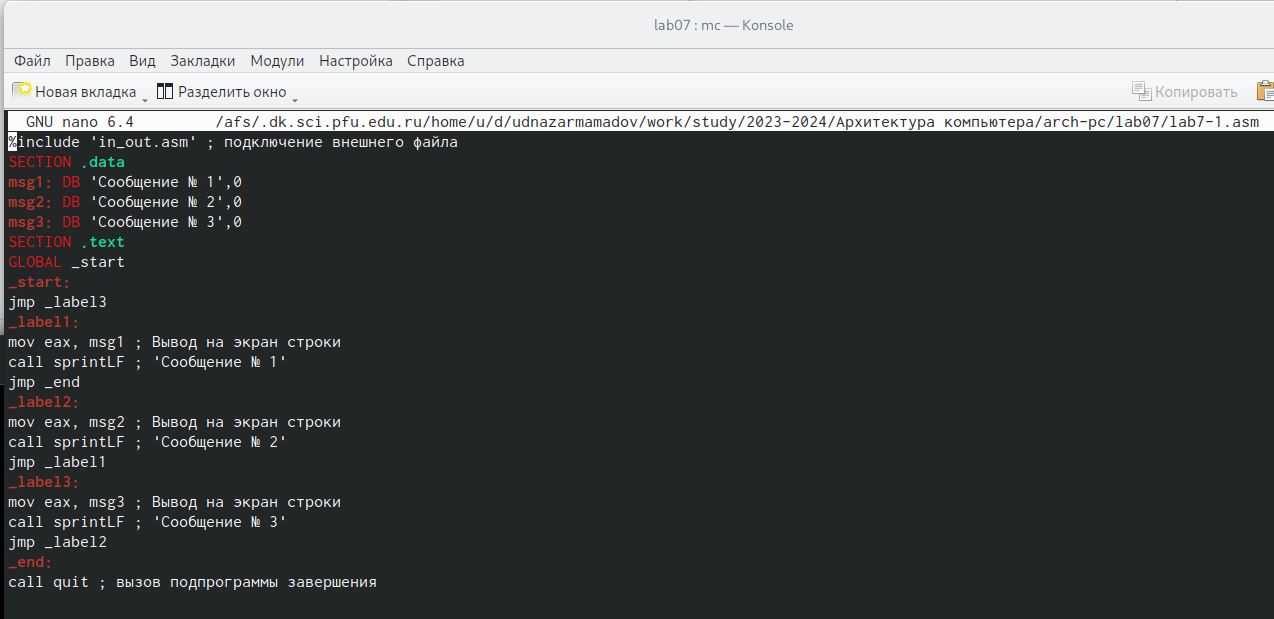
Редактирование файла

Создаю исполняемый файл и проверяю его работу (рис. -??).

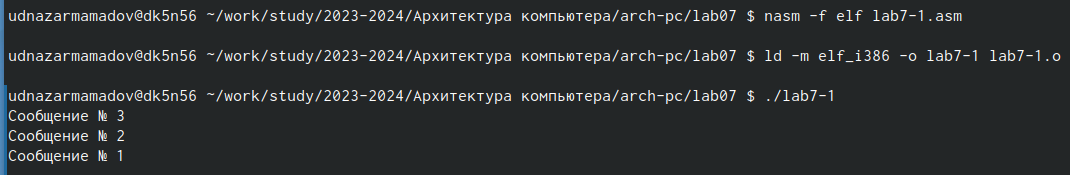


Запуск файла

Изменяю текст программы, добавив в начале программы jmp \_label3, jmp \_label2 в конце метки jmp \_label3, jmp \_label1 добавляю в конце метки jmp \_label2, и добавляю jmp \_end в конце метки jmp \_label1 (рис. -??).



Редактирование файла



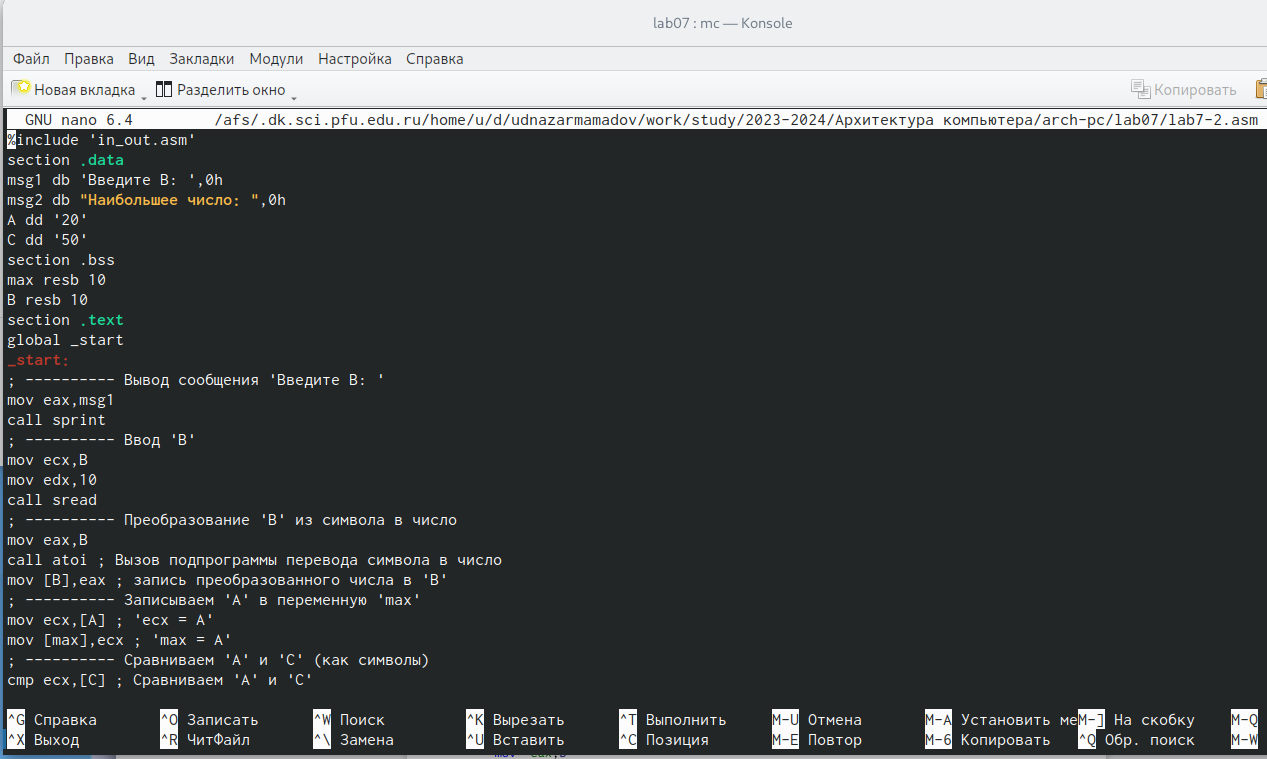
Редактирование файла

Рассмотрим программу, которая определяет и выводит на экран наибольшую из 3 целочисленных переменных: A,B и C. Значения для A и C задаются в программе, значение B вводиться с клавиатуры. Создаю файл lab7-2.asm в каталоге ~/work/arch-pc/lab07 (рис. -??).

Создание файла

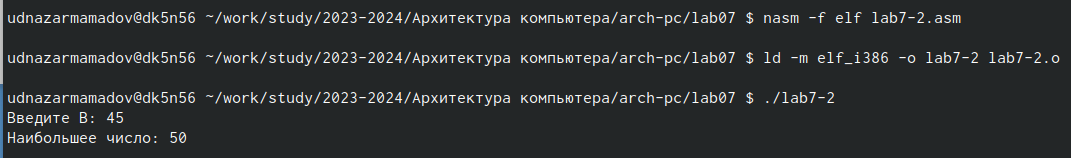
Создание файла

Вставляю текст программы из листинга 7.3 ввожу в lab7-2.asm (рис. -??).



Изменение файла

Создаю исполняемый файл и запускаю его (рис. -??).



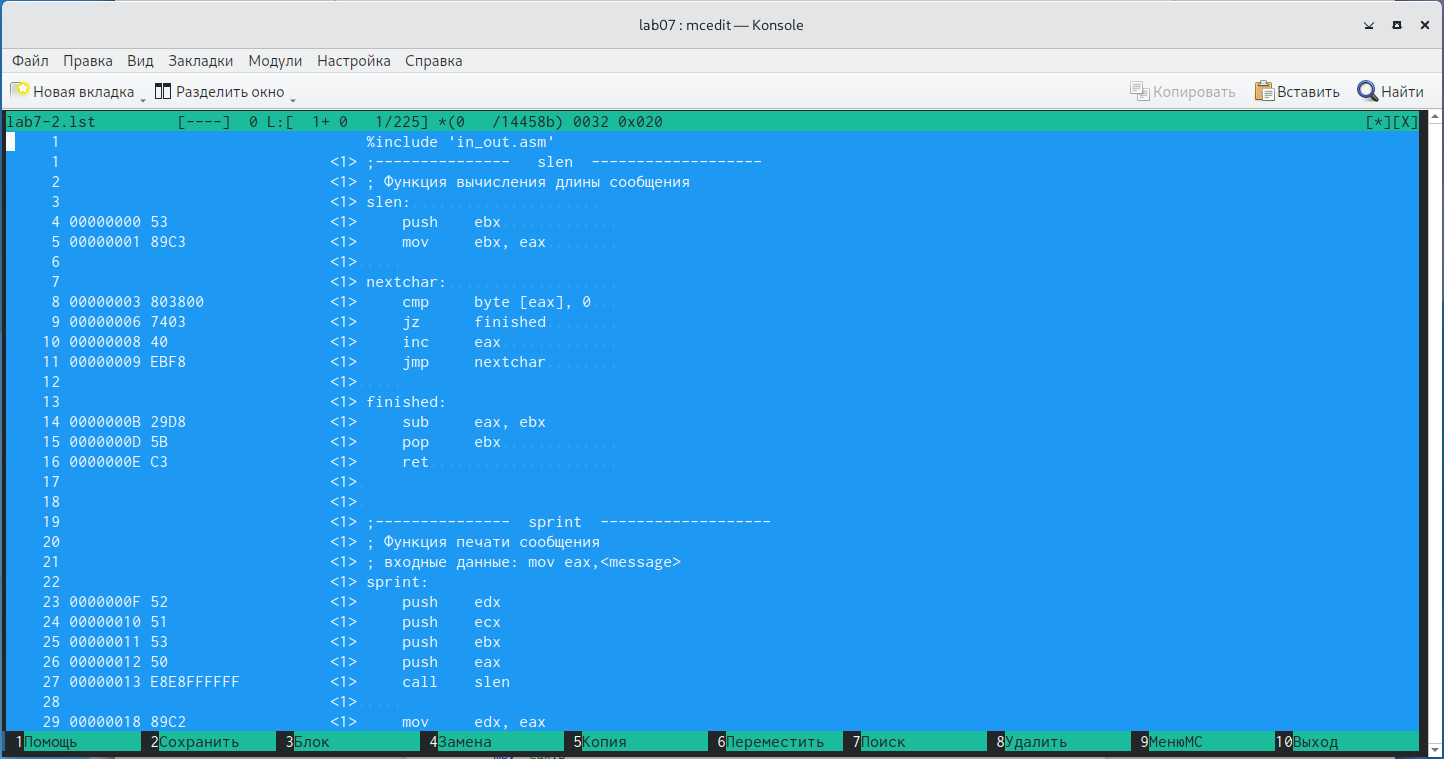
Запуск файла

Создаю файл листинга для программы из файла lab7-2.asm (рис. -??).

Создание файла

Создание файла

Открываю файл листинга lab7-2.lst с помощью текстового редактора и внимательно изучаю его формат и содержимое (рис. -??).



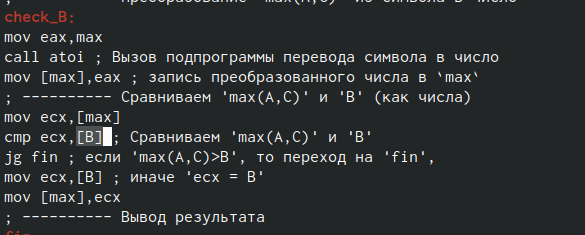
Изучение файла

В представленных трех строчках содержаться следующие данные (рис. -??).

Выбранные строки

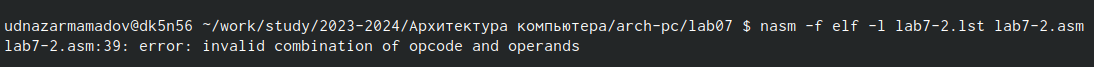
Выбранные строки

2” - номер строки кода, “; Функция вычисления длинны сообщения” “3” - номер строки кода, “slen” - название функции, не имеет адреса и машинного кода. “4” - номер строки кода, “00000000” - адрес строки, “53” - машинный код, “push ebx” - исходный текст программы, инструкция “push” помещает операнд “ebx” в стек. Открываю файл с программой lab7-2.asm и в выбранной мной инструкции с двумя операндами удаляю выделенный операнд (рис. -??).



Удаление выделенного операнда

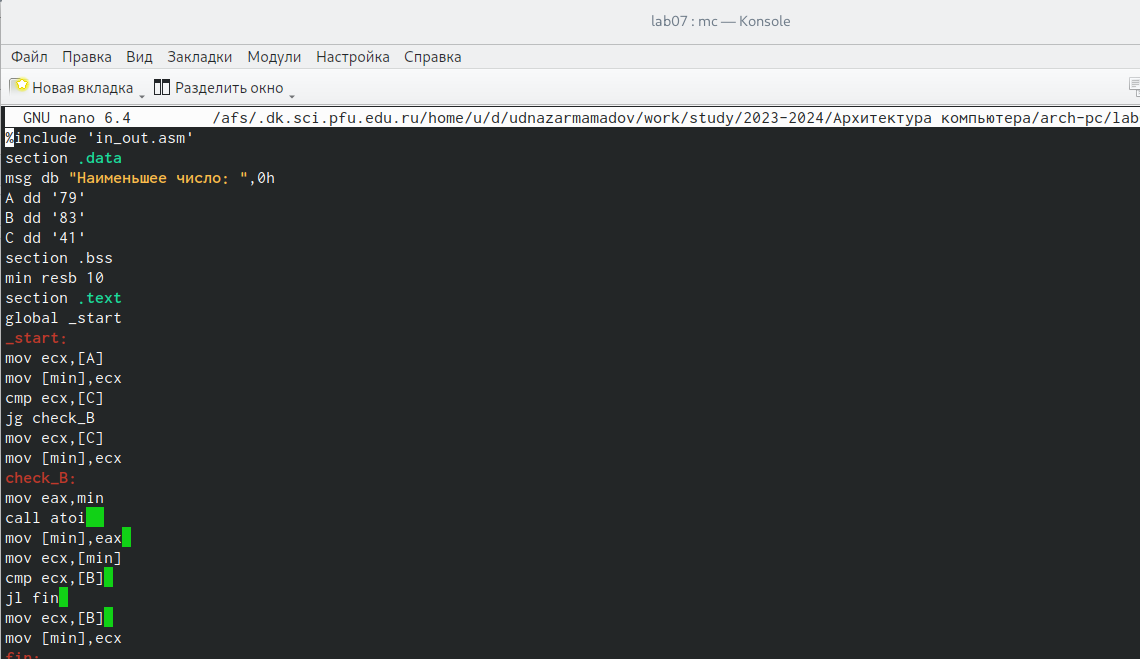
Выполняю трансляцию с получением файла листинга. (рис. -??). На выходе я получаю ошибку:инструкция mov (единственная в коде содержит два операнда) не может работать, имея только один операнд, из-за чего нарушается работа кода.



Запуск файла

#Задания для самостоятельной работы

1. Пишу программу нахождения наименьшей из 3 целочисленных переменных a, b и c. Значения переменных выбираю из табл. 7.5 в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Мой вариант под номером 6, поэтому мои значения - 79, 83 и 41. (рис. -??).



Решение задачи

Код программы:

%include ‘in\_out.asm’

section .data msg db “Наименьшее число:”,0h

A dd ‘79’

B dd ‘83’

C dd ‘41’

section .bss

min resb 10

section .text

global \_start

\_start:

mov ecx,[A]

mov [min],ecx

cmp ecx,[C] jg

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, msg

call sprint

mov eax,[min]

call iprintLF

call quit

# 5 Выводы

При выполнение данной лабораторной работы я изучил команды условного и безусловного переходов, приобрёл навыки написания программ с использованием переходов и ознакомилась с назначением и структурой файла листинга.

# Список литературы

1. GDB: The GNU Project Debugger. — URL: https://www.gnu.org/software/gdb/.
2. GNU Bash Manual. — 2016. — URL: https://www.gnu.org/software/bash/manual/.
3. Midnight Commander Development Center. — 2021. — URL: https://midnight-commander. org/.
4. NASM Assembly Language Tutorials. — 2021. — URL: https://asmtutor.com/.
5. Newham C. Learning the bash Shell: Unix Shell Programming. — O’Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: http://www.amazon.com/Learning- bash-Shell-Programming-Nutshell/dp/0596009658.
6. Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: https://www.nasm.us/docs.php.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс,
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: https://www.opennet.ru/docs/RUS/nasm/.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: http://www.stolyarov.info/books/asm\_unix.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер,
17. — 1120 с. — (Классика Computer Science).