title	subtitle	author	lang	toc-title	bibliography	csl	toc	toc-depth	lof
Отчёт по лабораторной работе №8	Простейший вариант	Волчкова Елизавета Дмитриевна	ru-RU	Содержание	bib/cite.bib	pandoc/csl/gost-r-7-0-5-2008-numeric.csl	true	2	true

Цель работы

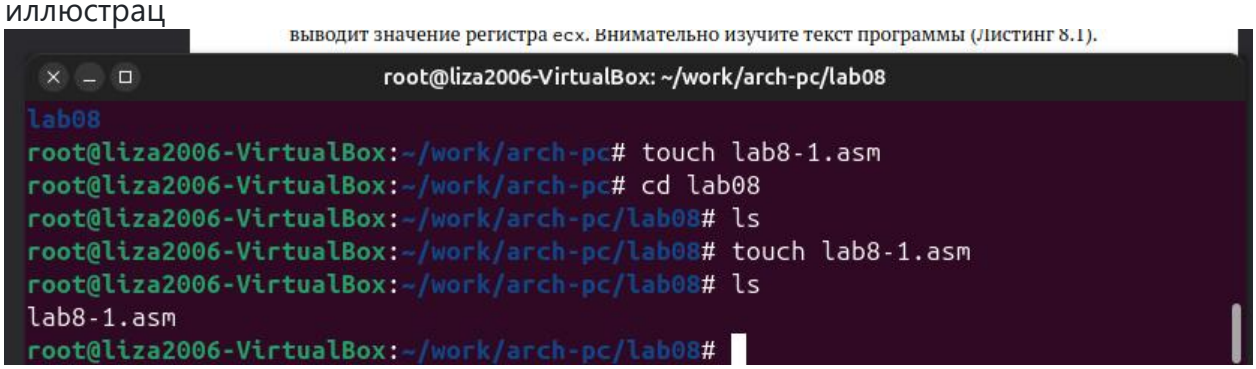
Приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

Задание

Напишите программу, которая находит сумму значений функции $f(x)$ для $x = x1$, $x2$, ..., xn, т.е. программа должна выводить значение $f(x1) + f(x2) + ... + f(xn)$. Значения xi передаются как аргументы. Вид функции $f(x)$ выбрать из таблицы 8.1 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу на нескольких наборах $x = x1$, $x2$, ..., xn.

Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрац

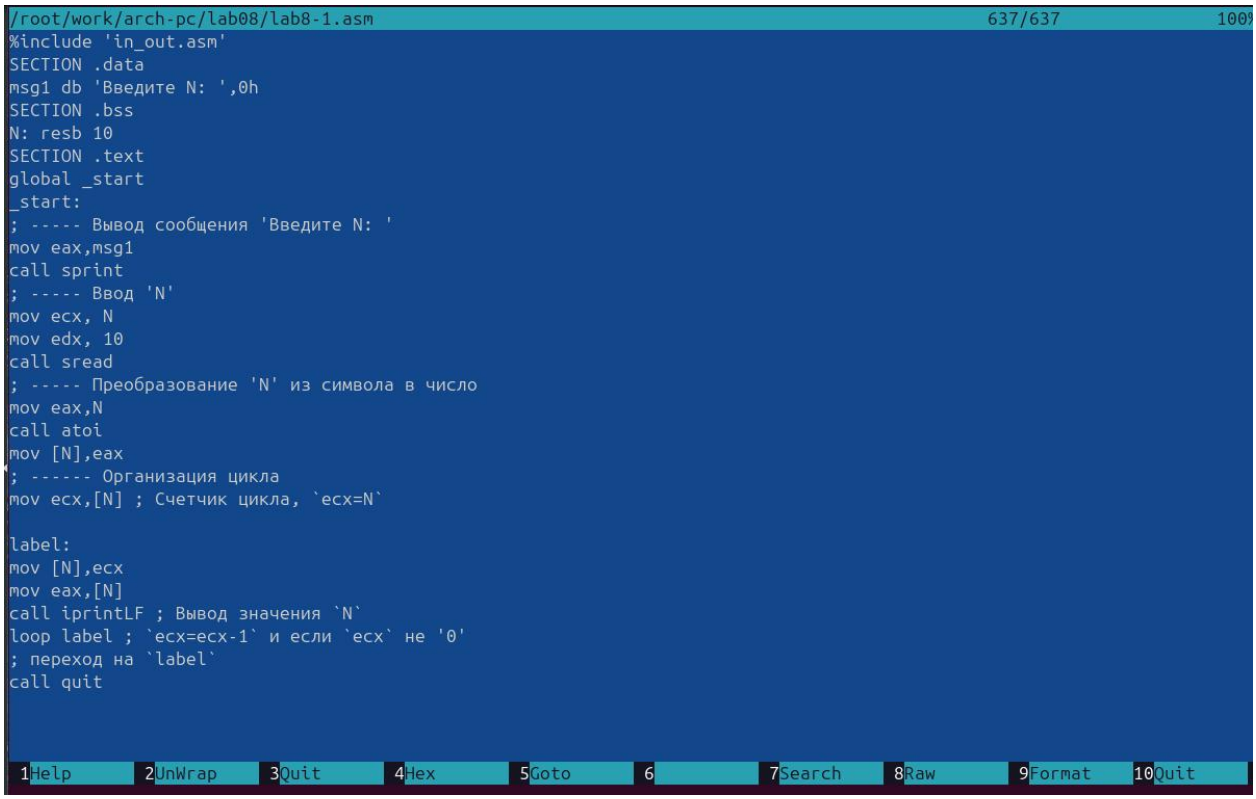


ию

текст программы из листинга 8.1. Затем создала исполняемый файл и проверила его работу.

Данный пример показывает, что использование регистра ecx в теле цилка loop может привести к некорректной работе программы. Далее я изменила текст программы добавив изменение значение регистра ecx в цикле.

Потом создала исполняемый файл и проверила его работу.



Какие значения принимает регистр ecx в цикле? Соответствует ли число проходов цикла значению N введенному с клавиатуры?

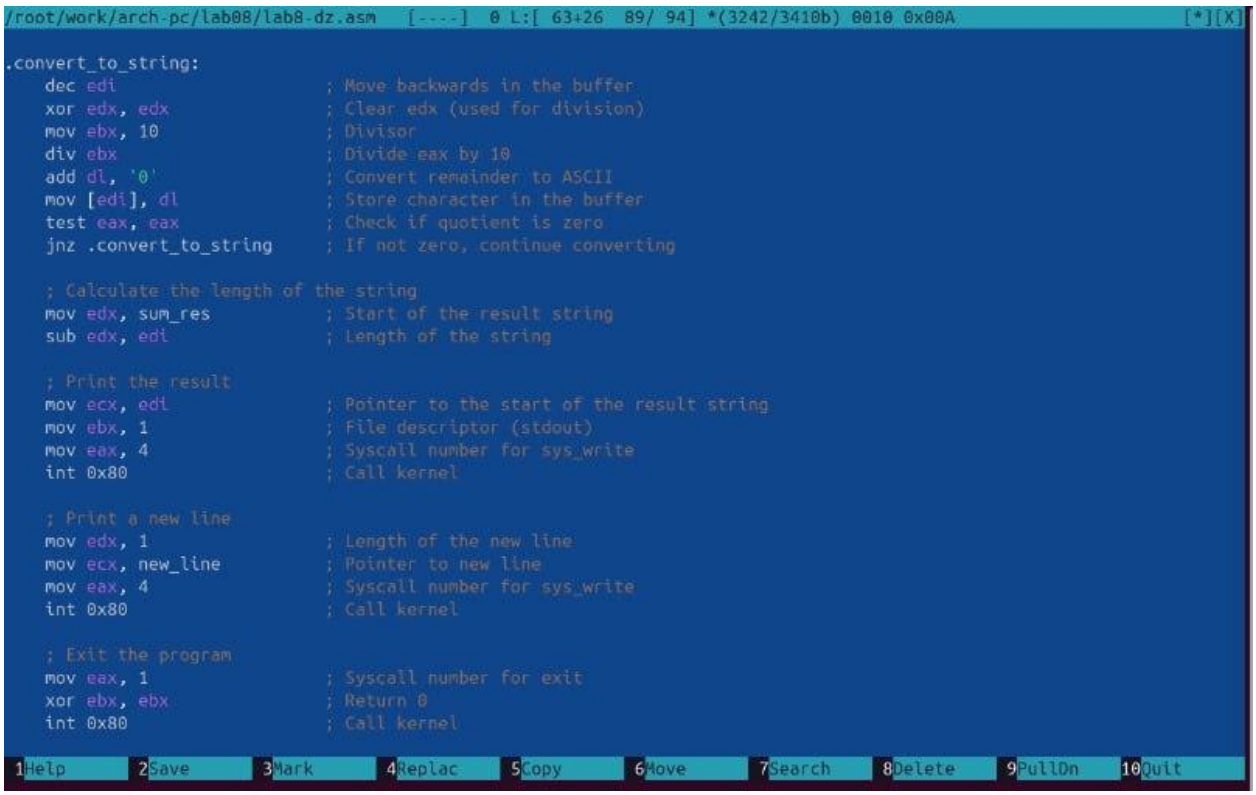
После внесла изменения в текст программы, добавив команды push и pop (добавления в стек и извлечения из стека) для сохранения значения счетчика цикла loop. Создала исполняемый файл и проверила его работу. Соответствует ли в данном случае число проходов цикла значению N введенному с клавиатуры? - нет! Циклов было больше 100.

Далее создала файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 и ввела в него текст про- граммы из листинга 8.2. Потом создала исполняемый файл и запустилаего, указав аргументы: Cколько аргументов было обработано программой?-

Затем создала файл lab8-3.asm в каталоге ~/work/arch-pc/lab08, ввела в него текст программы из листинга 8.3.

Я создала исполняемый файл и запустила его, указав аргументы.

Написала программу, которая находит сумму значений функции $f(x)$ для $x = x1, x2, ..., xn$, т.е. программа должна выводить значение $f(x1) + f(x2) + ... + f(xn)$. Значения xi передаются как аргументы. Вид функции $f(x)$ выбрала из таблицы 8.1 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7 и создалв исполняемый файл и проверилпа его работу на нескольких наборах $x = x1, x2, ..., xn$



Выводы

Целью работы было приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки, проделав данные задания я усвоила материал, имею представления о том, как правильно использовать циклы и командные строки.

Список литературы{.unnumbered}

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 c. — (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 c. — ISBN 978-1491941591. 7. The NASM documentation. — 2021. — URL: https://www.nasm.us/docs.php. 8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 c. — 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, 2017. 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, 2015. — 1120 с. — (Классика Computer Science). ::: {#refs} :::