Отчет по лабораторной работе №9

Дисциплина: архитектура компьютера

Сокирка Анна Константиновна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Приобретение навыков написания программ с использованием подпрограмм. Знакомство с методами отладки при помощи GDB и его основными возможностями.

2 Задание

1. Реализация подпрограмм в NASM
2. Отладка программ с помощью GDB
3. Самостоятельное выполнение заданий по материалам лабораторной работы

3 Теоретическое введение

Отладка — это процесс поиска и исправления ошибок в программе. В общем случае его можно разделить на четыре этапа: • обнаружение ошибки;
• поиск её местонахождения;
• определение причины ошибки;
• исправление ошибки.
Можно выделить следующие типы ошибок: • синтаксические ошибки — обнаруживаются во время трансляции исходного кода и вызваны нарушением ожидаемой формы или структуры языка;
• семантические ошибки — являются логическими и приводят к тому, что программа запускается, отрабатывает, но не даёт желаемого результата;
• ошибки в процессе выполнения — не обнаруживаются при трансляции и вызывают прерывание выполнения программы (например, это ошибки, связанные с переполнением или делением на ноль).
Второй этап — поиск местонахождения ошибки. Некоторые ошибки обнаружить довольно трудно. Лучший способ найти место в программе, где находится ошибка, это разбить программу на части и произвести их отладку отдельно друг от друга.
Третий этап — выяснение причины ошибки. После определения местонахождения ошибки обычно проще определить причину неправильной работы программы. Последний этап — исправление ошибки. После этого при повторном запуске программы, может обнаружиться следующая ошибка, и процесс отладки начнётся заново.

4 Выполнение лабораторной работы

4.1 4.1 Релазиация подпрограмм в NASM

Создаю каталог для выполнения лабораторной работы №9. Перейду в него и создам файл lab09-1.asm (рис. 4.1).

Рис. 4.1: Создание каталога

Копирую в файл код из листинга, компилирую и запускаю его, данная программа выполняет вычисление функции (рис. 4.2).

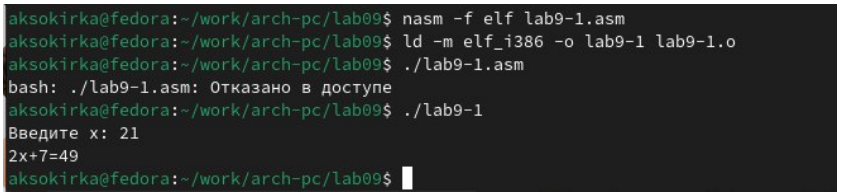

Рис. 4.2: Запуск файла

Изменяю текст программы, добавив в нее подпрограмму, теперь она вычисляет значение функции для выражения f(g(x)) (рис. 4.3).

Рис. 4.3: Изменение текста программы

4.2 4.1.1 Отладка программ с помощью GDB

В созданный файл копирую программу второго листинга, транслирую с созданием файла листинга и отладки, компоную и запускаю в отладчике (рис. 4.4).

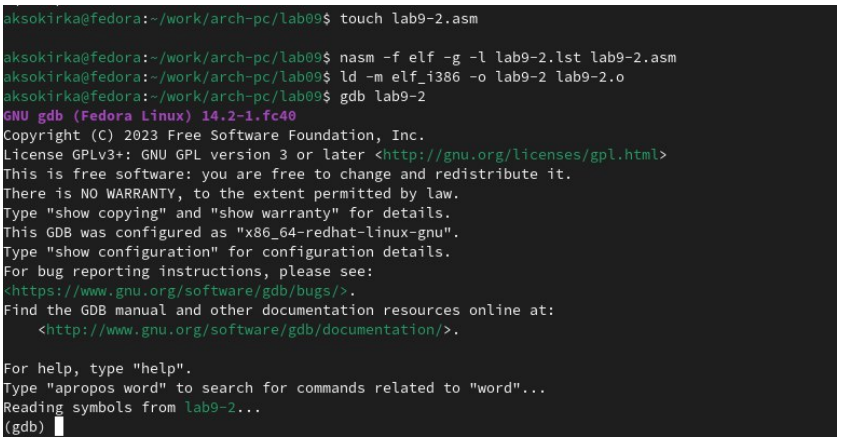

Рис. 4.4: Запуск файла

Запустив программу командой run, я убедилась в том, что она работает исправно (рис. 4.5).

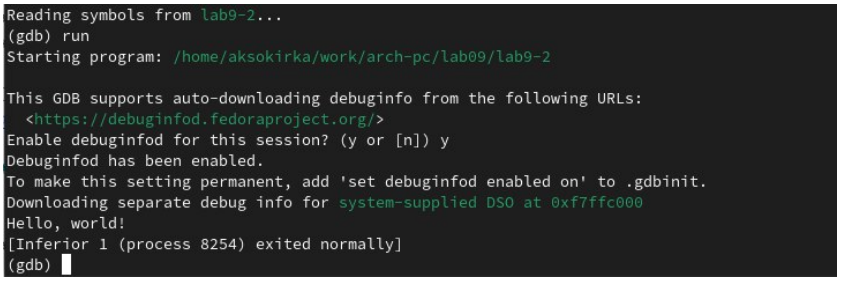

Рис. 4.5: Запуск программы

Для более подробного анализа программы добавляю брейкпоинт на метку _start и снова запускаю отладку (рис. 4.6).

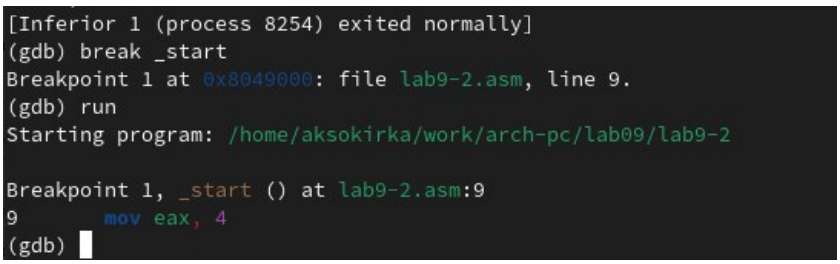

Рис. 4.6: Запуск программы

Смотрю дисассимилированный код программы с помощью команды disassemble начиная с метки _start(рис. 4.7).

Рис. 4.7: Просмотр кода программы с помощью метки

Переключаюсь на отображение команд с Intel'овским (рис. 4.8).

Рис. 4.8: Переключение

Различия между синтаксисом ATT и Intel заключаются в порядке операндов (ATT - Операнд источника указан первым. Intel - Операнд назначения указан первым), их размере (ATT - размер операндов указывается явно с помощью суффиксов, непосредственные операнды предваряются символом $; Intel - Размер операндов неявно определяется контекстом, как ax, eax, непосредственные операнды пишутся напрямую), именах регистров(ATT - имена регистров предваряются символом %, Intel - имена регистров пишутся без префиксов).

Включаю режим псевдографики для более удобного анализа программы (рис. 4.9).

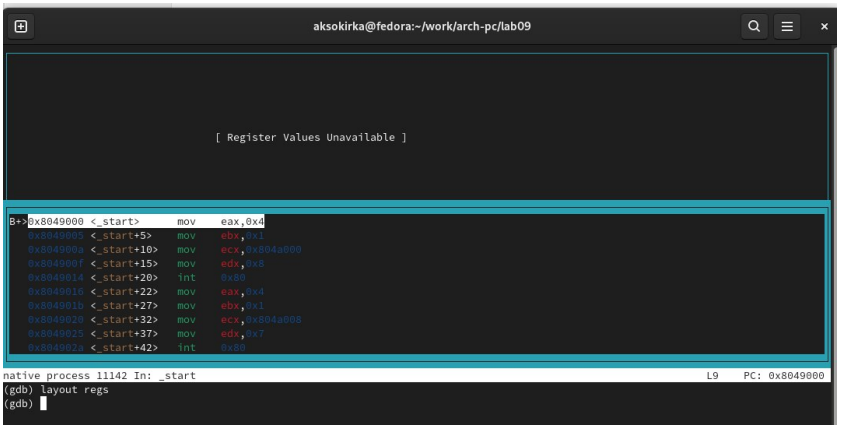

Рис. 4.9: Режим псевдографики

4.3 4.1.3 Работа с данными программы в GDB

Проверяем была ли установлена точка останова и устанавливаем точку останова предпоследней инструкции (рис. 4.10).

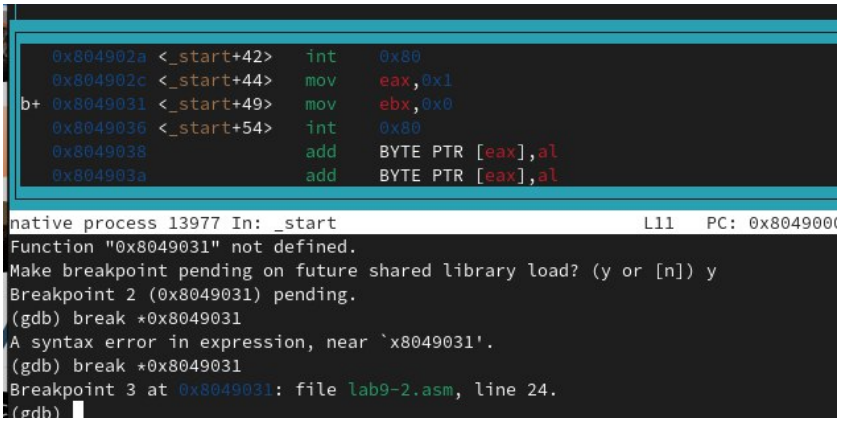

Рис. 4.10: Используем команду info breakpoints и создаем новую точку останова

Посмотрим информацию о всех установленных точках останова(рис. 4.11).

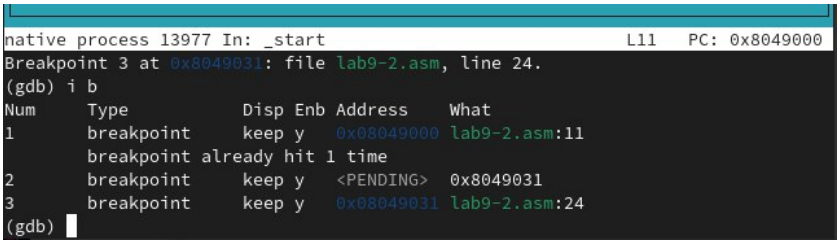

Рис. 4.11: Смотрим информацию

Выполняем 5 инструкций командой si (рис. 4.12).

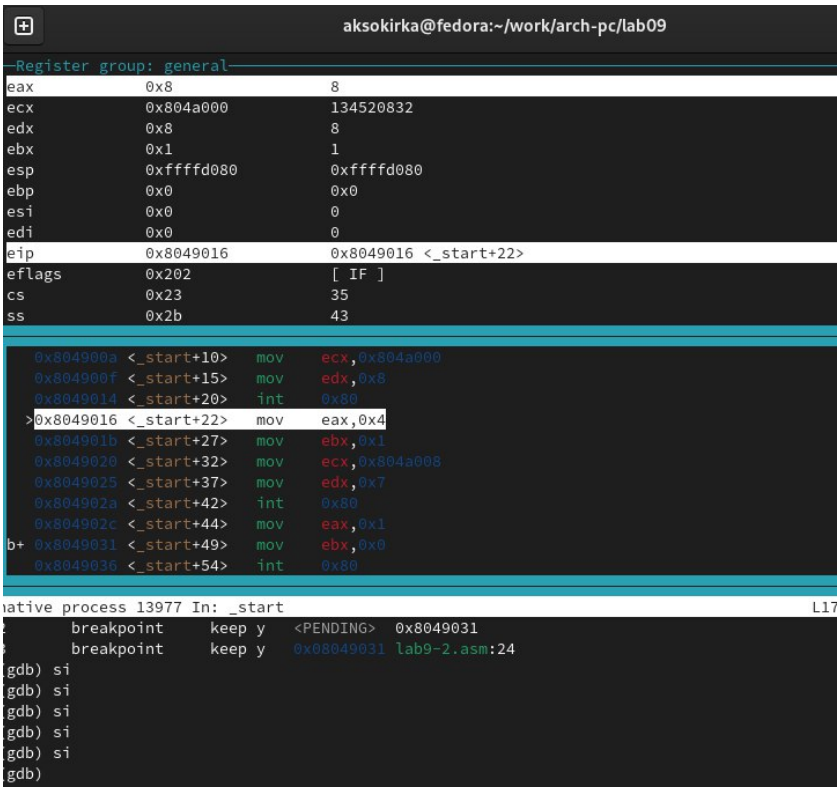

Рис. 4.12: Отслеживаем регистры

Во время выполнения команд менялись регистры: ebx, ecx, edx,eax, eip. Смотрим значение переменной msg1 по имени (рис. 4.13).

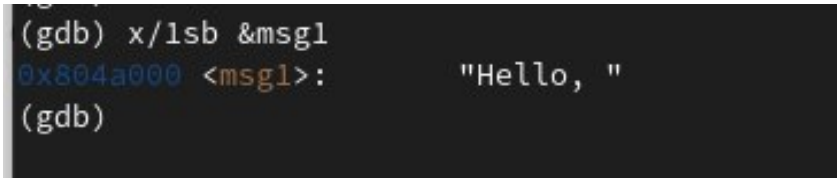

Рис. 4.13: Смотрим значение переменной

Смотрим значение переменной msg2 по адресу (рис. 4.14).

Рис. 4.14: Смотрим значение переменной

Изменим первый символ переменной msg1 (рис. 4.15).

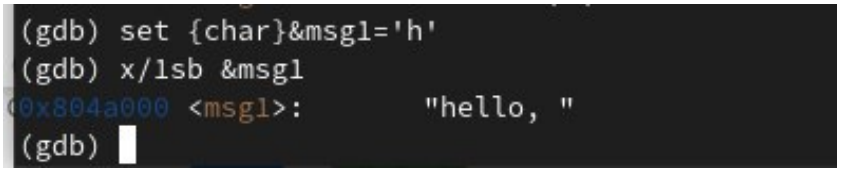

Рис. 4.15: Меняем символ

Изменим первый символ переменной msg2 (рис. 4.16).

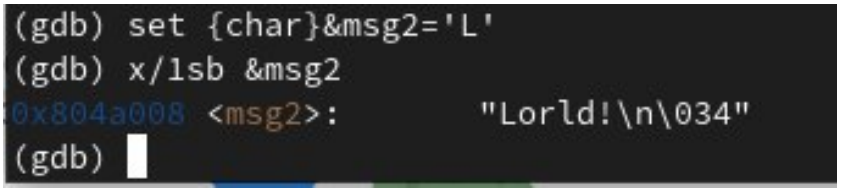

Рис. 4.16: Меняем символ

Смотрим значение регистра edx в разных форматах (рис. 4.17).

Рис. 4.17: Смотрим значение регистра

Изменяем регистор ebx (рис. 4.18).

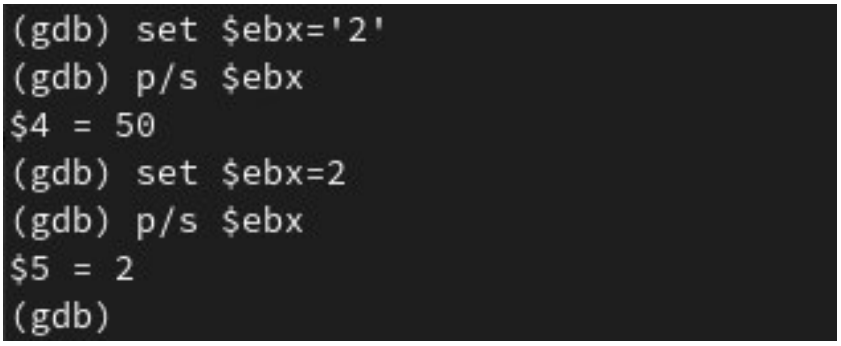

Рис. 4.18: Изменяем регистр командой set

Выводится разные значения, так как команда без кавычек присваивает регистру вводимое значение. Прописываем команды для завершения программы и выхода из GDB (рис. 4.19).

Рис. 4.19: Прописываем команды c и quit

4.4 4.1.4 Обработка аргументов командной строки в GDB

Копирую программу из предыдущей лабораторной работы в текущий каталог и и создаю исполняемый файл с файлом листинга и отладки (рис. 4.20).

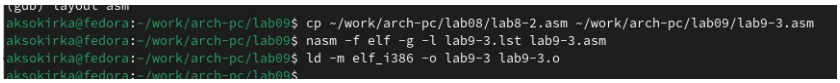

Рис. 4.20: Подготовка новой программы

Запускаем его в отладчике GDB. Установим точку останова перед первой инструкцией в программе и запустим ее (рис. 4.21).

Устанавливаем точку останова

Рис. 4.21: Устанавливаем точку останова

Смотрим позиции стека по разным адресам (рис. 4.22).

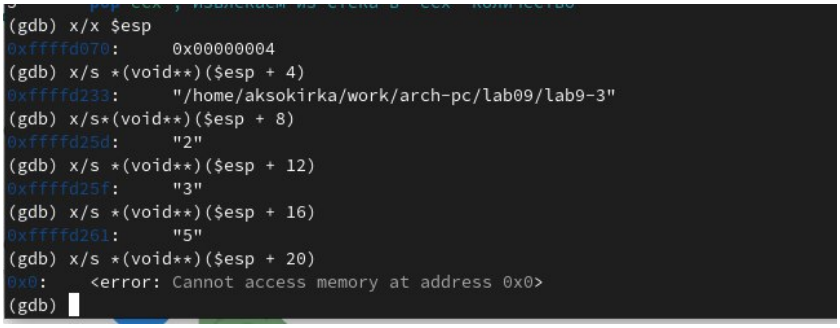

Рис. 4.22: Изучаем полученные данные

Шаг изменения адреса равен 4 потому что адресные регистры имеют размерность 32 бита(4 байта).

4.5 4.2 Задание для самостоятельной работы

Создаем новый файл в директории.Открываем файл в Midnight Commander и заполняем его в соответствии с листингом 9.3. Создаем исполняемый файл и

запускаем его (рис. 4.23).

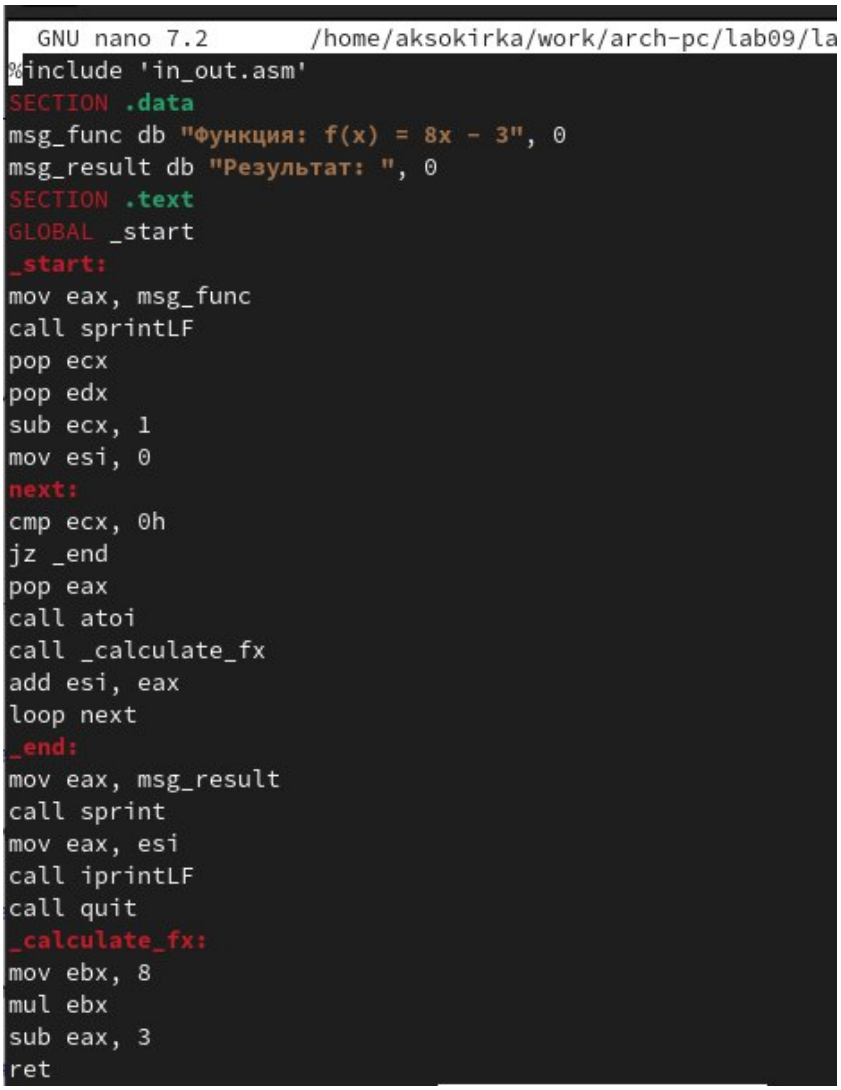

Рис. 4.23: Создаем файл

5 Выводы

В результате выполнения данной лабораторной работы я приобрела навыки написания программ с использованием подпрограмм, а так же познакомилась с методами отладки при помощи GDB и его основными возможностями.

6 Список литературы

1 {#https://esystem.rudn.ru/pluginfile.php/2089096/mod_resource/content/0/Лабораторная%20работа%20№9.%20Понятие%20подпрограммы.%20Отладчик%20..pdf}