Лабораторная работа №9. Понятие подпрограммы. Отладчик GDB.

НПМбв-01-21

Ермаков Алексей

Содержание

# Цель работы

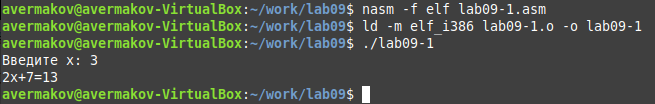
Получить навыки написания программ с использованием подпрограмм. Познакомиться с методами отладки при помощи GDB и его основными возможностями.

# Задание

1. Преобразуйте программу из лабораторной работы №8 (Задание №1 для самостоятель- ной работы), реализовав вычисление значения функции 𝑓(𝑥) как подпрограмму.
2. В листинге 9.3 приведена программа вычисления выражения (3 + 2) ∗ 4 + 5. При запуске данная программа дает неверный результат. Проверьте это. С помощью отладчика GDB, анализируя изменения значений регистров, определите ошибку и исправьте ее

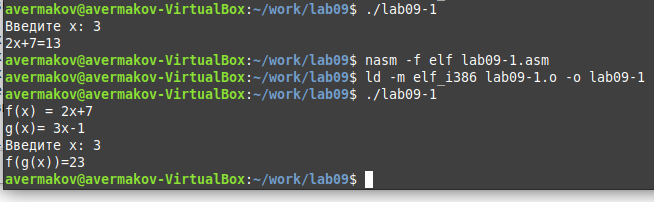
# Выполнение лабораторной работы

Создал каталог для программам лабораторной работы, написал в файл lab09-1.asm текст программы из листинга 9.1. Создал исполняемый файл и проверил его работу. (рис. @fig:001).



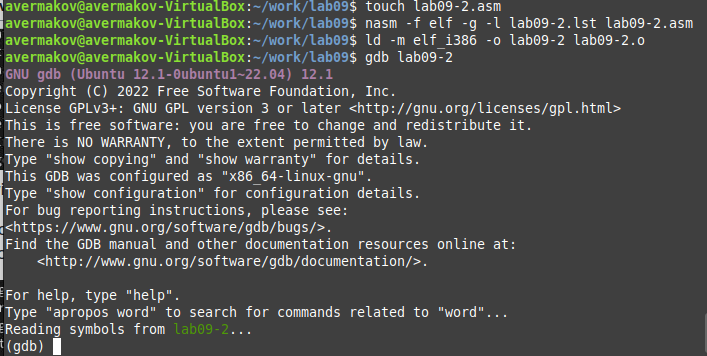
Создание и проверка работы файла вычисления арифметического выражения

Изменил текст программы добавив добавив две подпрограммы вычисления арифметического выражения. Создал исполняемый файл и проверил его работу(рис. @fig:002).



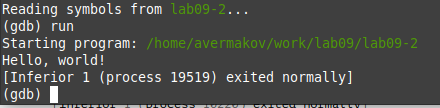
Пример работы файла с двумя подпрограммами вычисления арифметического выражения

Создал файл lab09-2.asm с текстом программы из Листинга 9.2. (Программа печати сообщения Hello world!) и проверил его работу. Начал отладку(рис. @fig:003).



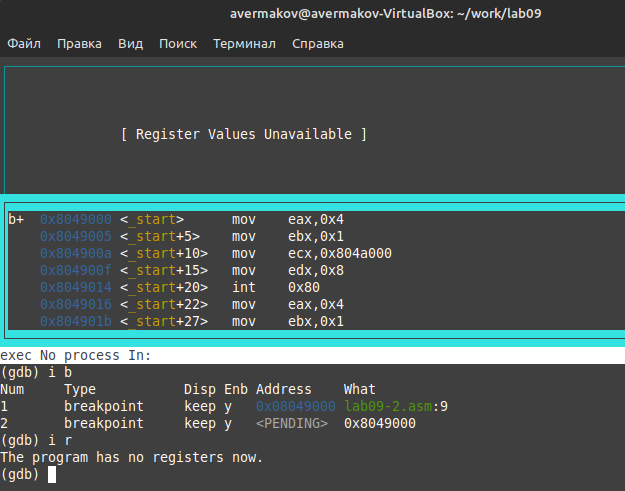
Создание и проверка работы файла печати сообщения и запуск отладки

Проверил работу программы, запустив ее в оболочке GDB с помощью команды run. (рис. @fig:004).



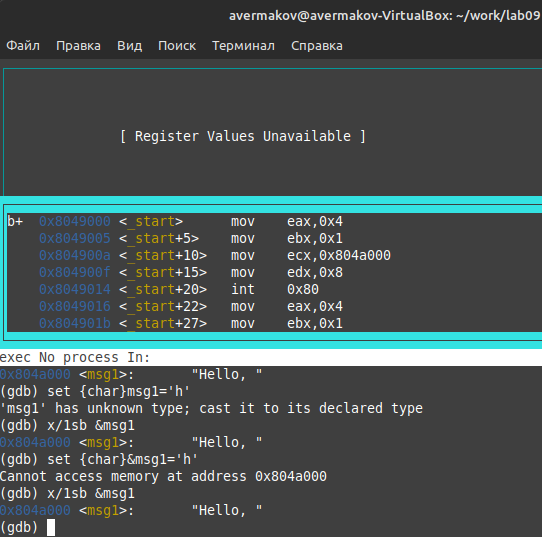
Создание и проверка работы файла вычисления арифметического выражения

В режиме псевдографики gdb была установлена точка останова по адресу инструкции. (рис. @fig:005).



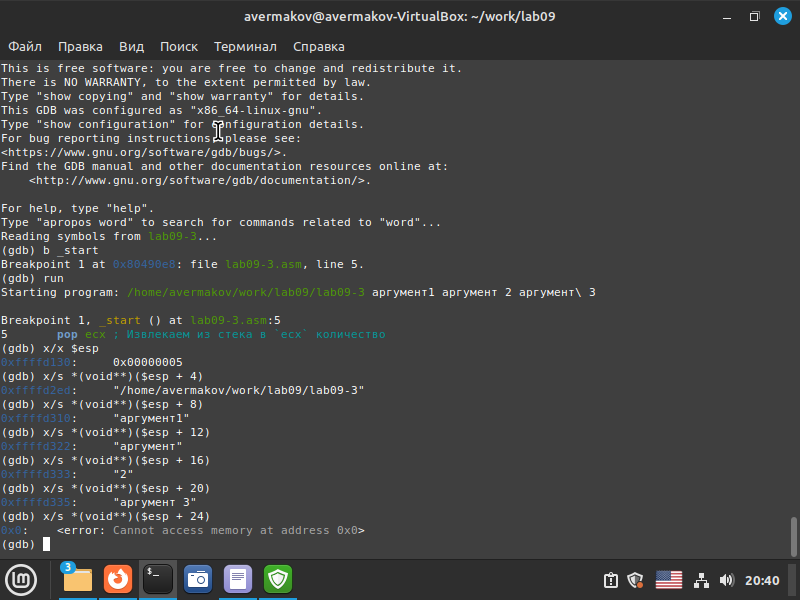
Установил точка останова по адресу инструкции в режиме псевдографики

Посмотрели значение переменной по адресу используя отображения содержимого памяти. Посмотрели инструкцию mov ecx,msg2 которая записывает в регистр ecx адрес перемененной msg2 (рис. @fig:006).



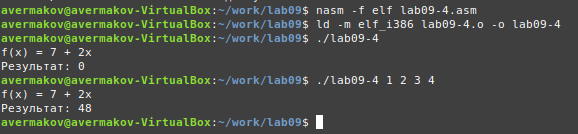
Посмотр значение переменной по адресу используя отображения содержимого памяти

Посмотр позиции стека. Размер переменной - четыре байта и шаг изменения адреса равен размеру переменной (рис. @fig:007).



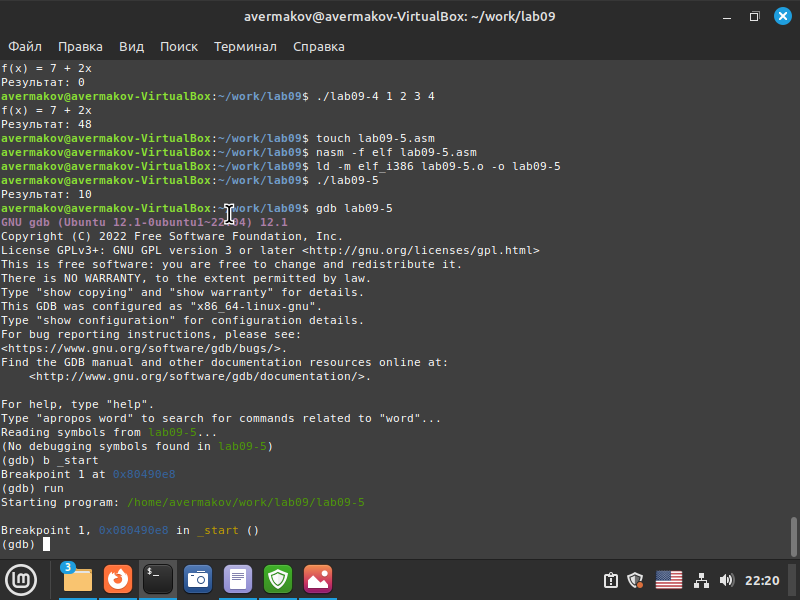
Посмотр позиции стека и определение размера шага изменения адреса.

Преобразовали программу из лабораторной работы №8 (Задание №1 для самостоятельной работы), реализовав вычисление значения функции как подпрограмму. (рис. @fig:008).



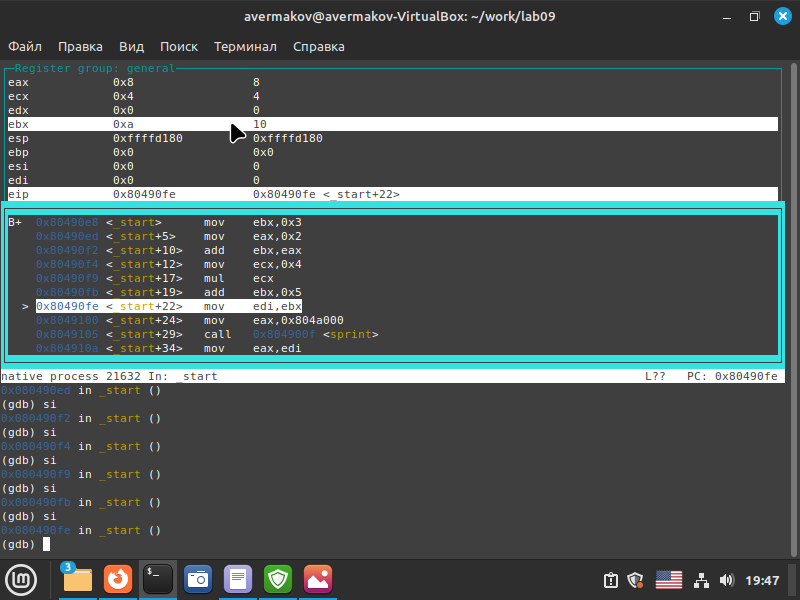
Преобразовали программу из лабораторной работы №8, реализовав вычисление значения функции как подпрограмму

Создали файл вычисления арифметического выражения, проверили его работу, обратили внимание на ошибку и запустили отладку. (рис. @fig:009).



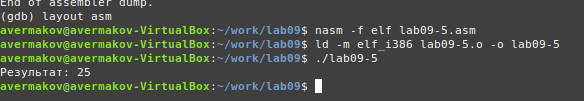
Создали файл вычисления арифметического выражения, проверили его работу, обратили внимание на ошибку и запустили отладку.

С помощью отладчика GDB, анализируя изменения значений регистров,определил ошибку и исправил(рис. @fig:0010).



Определение ошибки с помощью отладчика GDB

Создал исполняемый исправленный файл, проверил его работу и убедился, что работает скрипт корректно. (рис. @fig:0011).



Создание и проверка работы исправленного файла вычисления арифметического выражения

# Выводы

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