Отчёт по лабораторной работе №9

Дисциплина: Архитектура компьютера

Мария Данииловна Гольцова

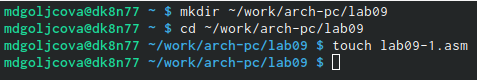
Содержание

# 1 Цель работы

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

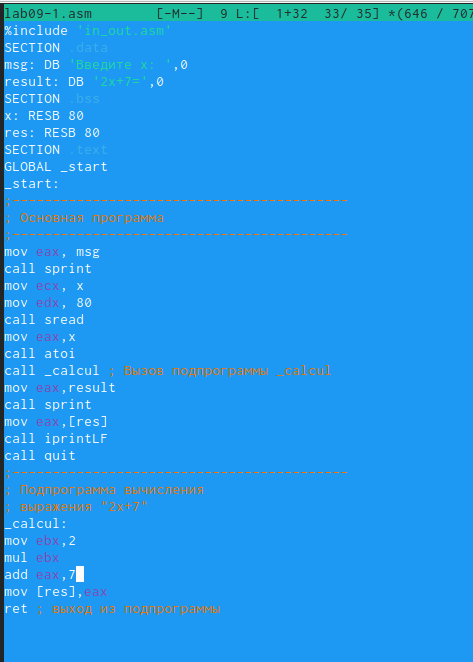
# 2 Выполнение лабораторной работы

Создала каталог для выполнения лабораторной работы №9, перешла в него и создала файл (рис. ??).



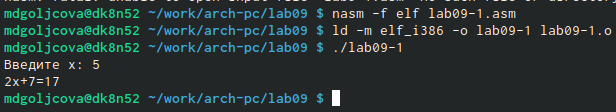
Создание каталога, файла

Ввела в файл lab09-1.asm текст программы из листинга 9.1 (рис. ??).



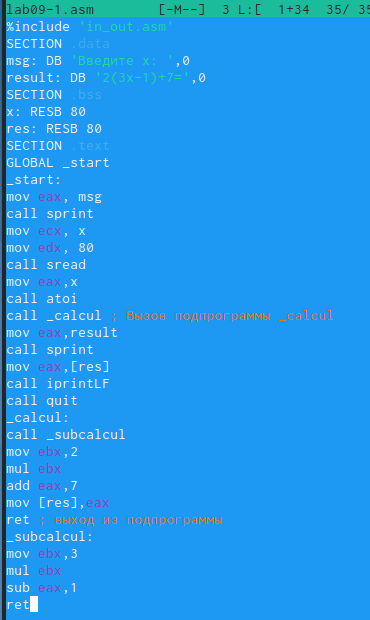
Текст программы 1

Создала исполняемый файл и проверила его работу (рис. ??).



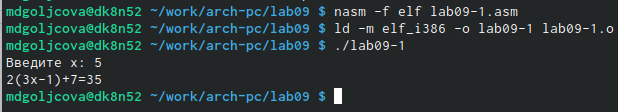
Работа программы 1

Изменила текст программы, добавив подпрограмму \_subcalcul в подпрограмму \_calcul, для вычисления выражения f(g(x)), где x вводится с клавиатуры, f(x)=2x+7, g(x)=3x-1 (рис. ??).



Изменение программы 1

Создала исполняемый файл и проверила его работу (рис. ??).



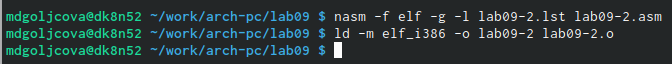
Работа программы

Создала файл lab09-2.asm с текстом программы из листинга 9.2 (рис. ??).



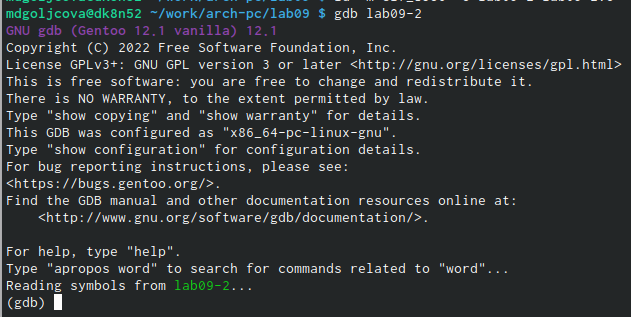
Текст программы 2

Получила исполняемый файл (рис. ??).



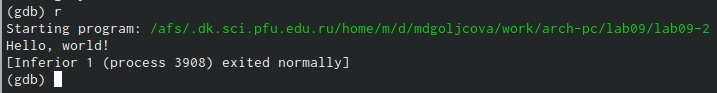
Исполняемый файл

Загрузила исполняемый файл в отладчик gdb (рис. ??).



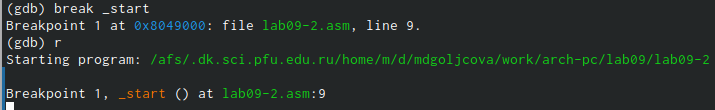
Загрузка файла в отладчик gdb

Проверила работу программы, запустив её в оболочке GDB с помощью команды run (рис. ??).



Проверка работы программы

Для более подробного анализа программы установила брейкпоинт на метку \_start, с которой начинается выполнение любой ассемблерной программы, и запустила её (рис. ??).



Установка брейкпоинта и запуск программы

Посмотрела дисассимилированный код программы с помощью команды disassemble, начиная с метки \_start (рис. ??).



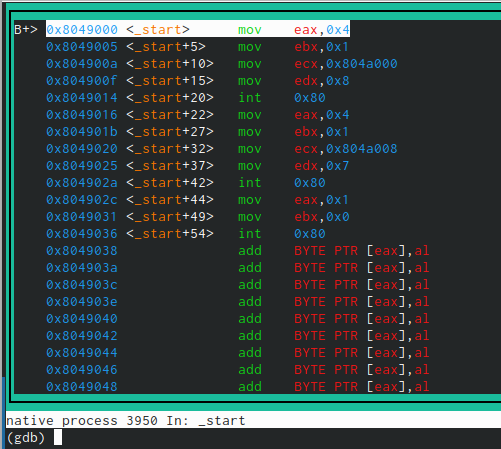
Дисассимилированный код программы

Переключение на отображение команд с Intelовским синтаксисом (рис. ??).



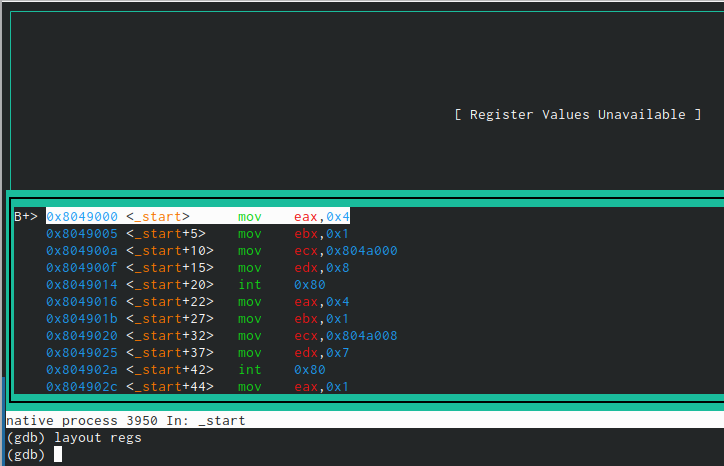
Переключение на отображение команд

Включила режим псевдографики для более удобного анализа программы: layout asm (рис. ??).



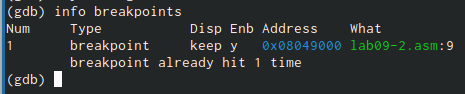
Режим псевдографики

layout regs (рис. ??).



Режим псевдографики

Проверила точку останова по имени метки (рис. ??).



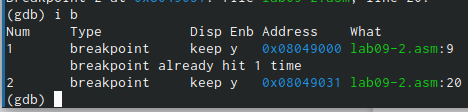
Точка останова

Определила адрес предпоследней инструкции (mov ebx,0x0) и установила точку останова (рис. ??).

Адрес предпоследней инструкции

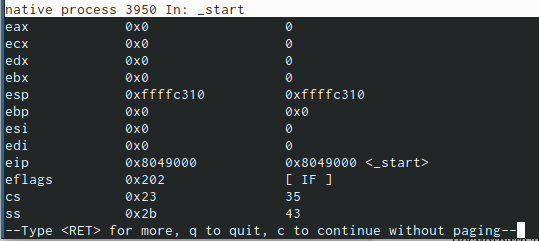
Адрес предпоследней инструкции

Посмотрела информацию о всех установленных точках останова (рис. ??).



Информация о всех установленных точках останова

Посмотрела содержимое регистров (рис. ??).



Содержимое регистров

Посмотрела значение переменной msg1 по имени (рис. ??).

Значение переменной 1

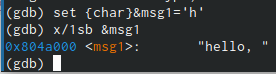
Значение переменной 1

Посмотрела значение переменной msg2 по адресу (рис. ??).

Значение переменной 2

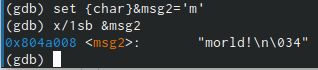
Значение переменной 2

Изменила первый символ переменной msg1 (рис. ??).



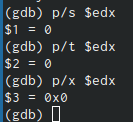
Изменение первого символа msg1

Заменила первый символ во второй переменной msg2 (рис. ??).



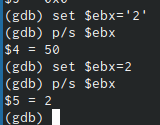
Изменение первого символа msg2

Вывела в различных форматах (в шестнадцатеричном формате, в двоичном формате и в символьном виде) значение регистра edx (рис. ??).



Вывод значений регистра edx

С помощью команды set изменила значение регистра ebx (рис. ??).



Изменение регистра ebx

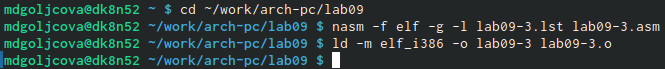
Завершила выполнение программы с помощью команды continue (сокращенно c) или stepi (сокращенно si) и вышла из GDB с помощью команды quit (сокращенно q).

Скопировала файл lab8-2.asm, созданный при выполнении лабораторной работы №8, с программой выводящей на экран аргументы командной строки (Листинг 8.2) в файл с именем lab09-3.asm (рис. ??).

Копирование

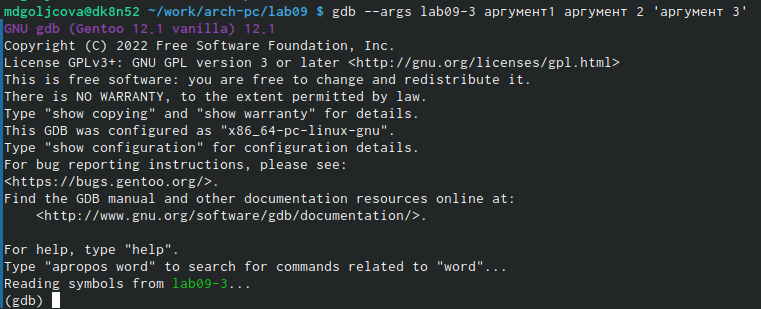
Копирование

Создала исполняемый файл (рис. ??).



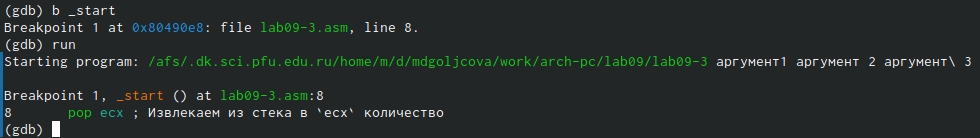
Создание исполняемого файла

Загрузила исполняемый файл в отладчик, указав аргументы (рис. ??).



Загрузка в отладчик

Для начала установлю точку останова перед первой инструкцией в программе и запущу ее (рис. ??).



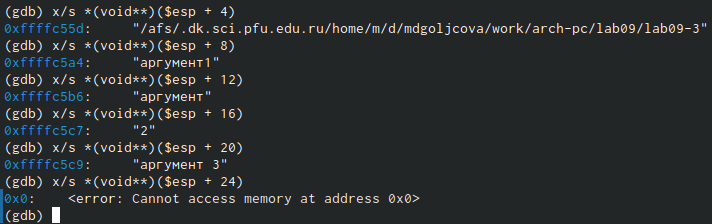
Установка точки останова

Адрес вершины стека храниться в регистре esp, и по этому адресу располагается число, равное количеству аргументов командной строки (включая имя программы) (рис. ??).

Адрес вершины стека

Адрес вершины стека

Посмотрела остальные позиции стека (рис. ??).



Позиции стека

Шаг изменения адреса равен 4, потому что число аргументов равно 4.

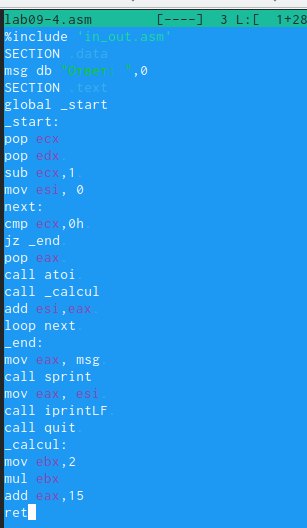
# 3 Самостоятельная работа

Создала файл для самостоятельной работы (рис. ??).

Новый файл

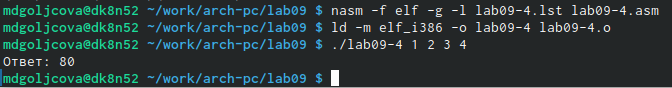
Новый файл

Преобразовала программу из лабораторной работы №8 (Задание №1 для самостоятельной работы), реализовав вычисление значения функции 𝑓(𝑥) как подпрограмму (рис. ??).



Преобразованная программа

Получила верный ответ (рис. ??).



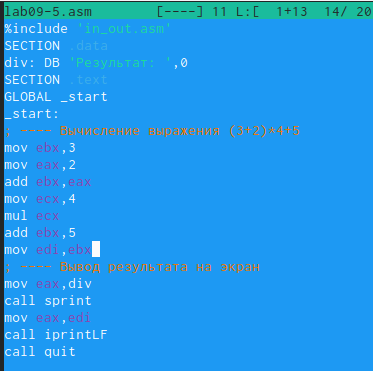
Результат

Создала файл для второго задания самостоятельной работы (рис. ??).

Новый файл

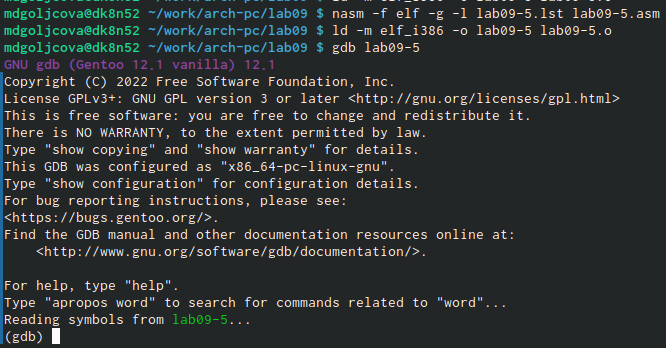
Новый файл

Ввела программу из листинга 9.3 (рис. ??).



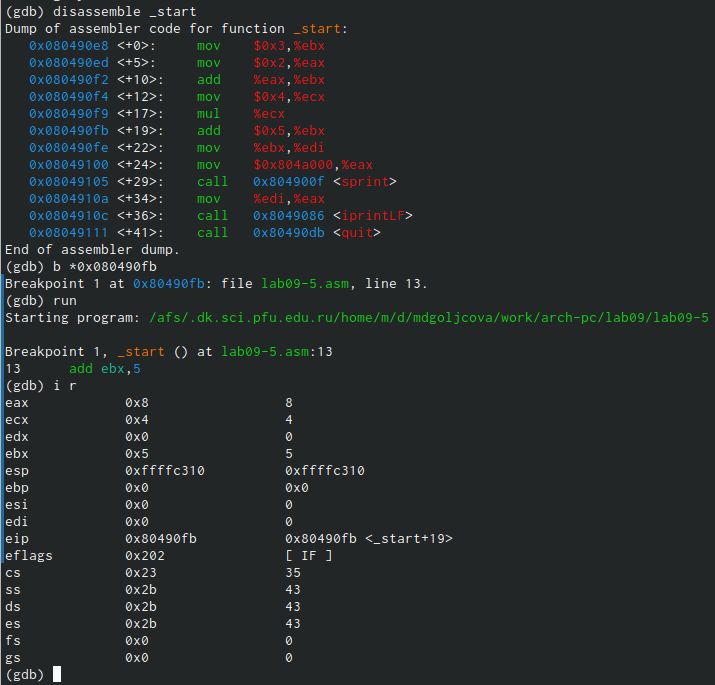
Текст программы

Попробовала запустить программу (рис. ??).



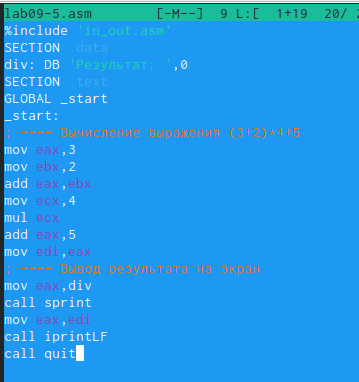
Запуск

Просматрела дисассимилированный код программы, поставила точку останова перед прибавлением 5 и открыла значения регистров на данном этапе (рис. ??).



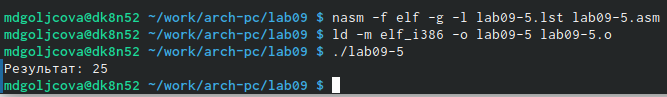
Дисассимилированный код, значения регистров

Регистр ecx со значением 4 умножается не на ebx,сложенным c eax, а только с eax со значением 2. Меняю (рис. ??).



Измененная программа

Получаю верный ответ (рис. ??).



Результат

# 4 Выводы

В ходе работы я приобрела навыки написания программ с использованием подпрограмм, познакомилась с методами отладки при помощи GDB и его основными возможностями.

# Список литературы