Отчёт по лабораторной работе №9

Понятие подпрограммы.Отладчик GBD

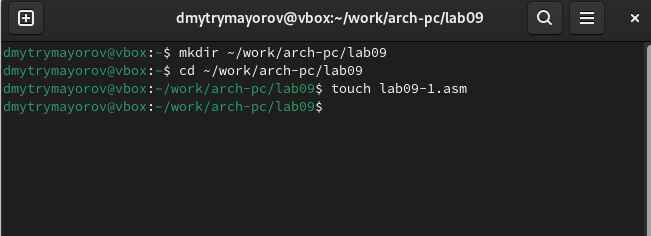
Майоров Дмитрий Андреевич

Содержание

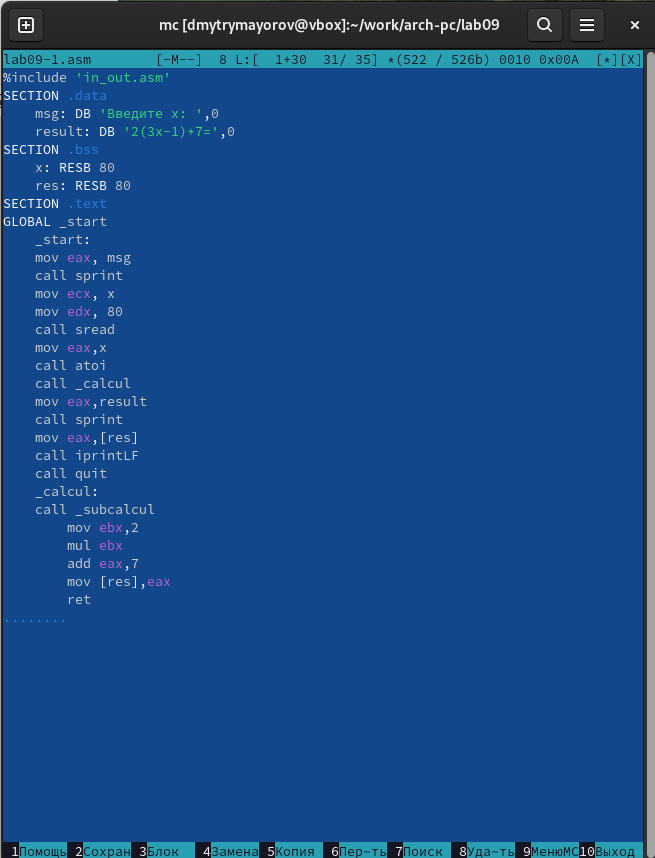
# 1 Цель работы

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

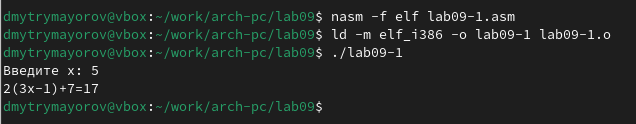
# 2 Выполнение лабораторной работы



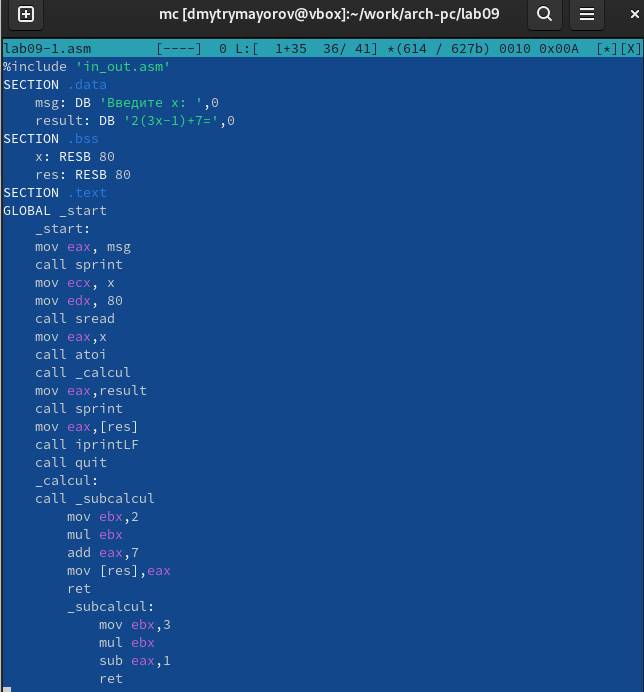
Создаем каталог для програм лабораторной работы и файл в нем



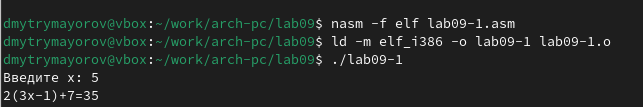
Открываем файл и заполняем его в соотвествии с листингом



Создаем исполняемый файл и запускаем его



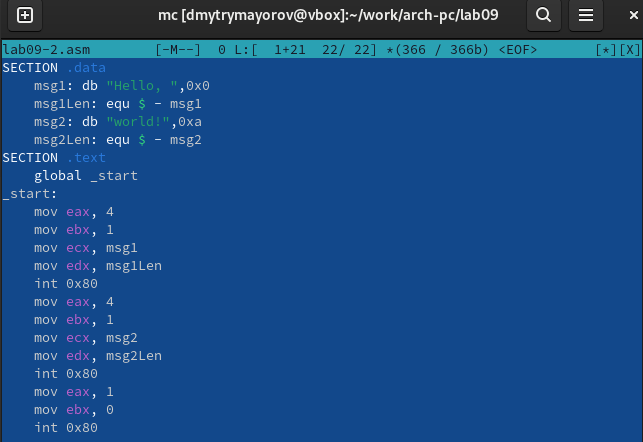
Изменяем файл. Добавляем подпрограмму



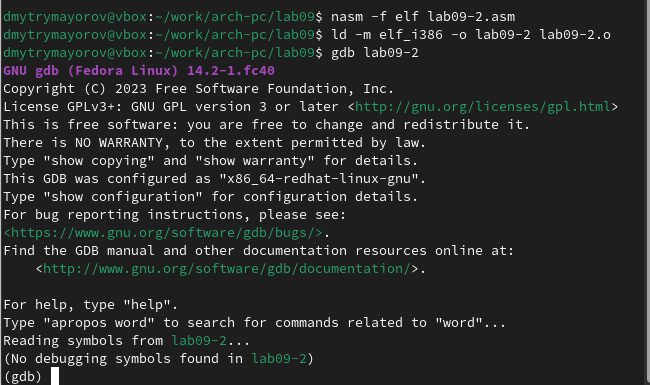
Создаем исполняемый файл и запускаем его

Создаем новый файл

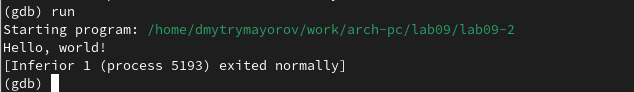
Создаем новый файл



Открываем файл и заполняем его в соотвествии с листингом



Загружаем исходный файл в отладчик



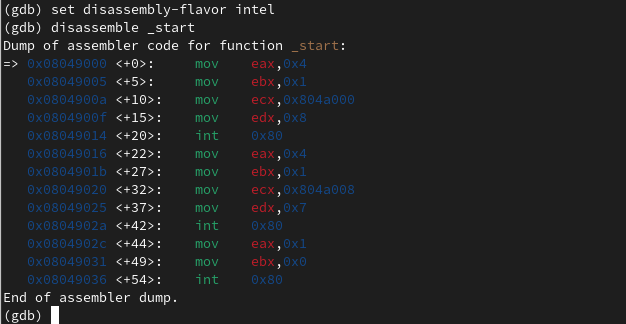
Запускаем программу командой run



Запускаем программу с брейкпоином

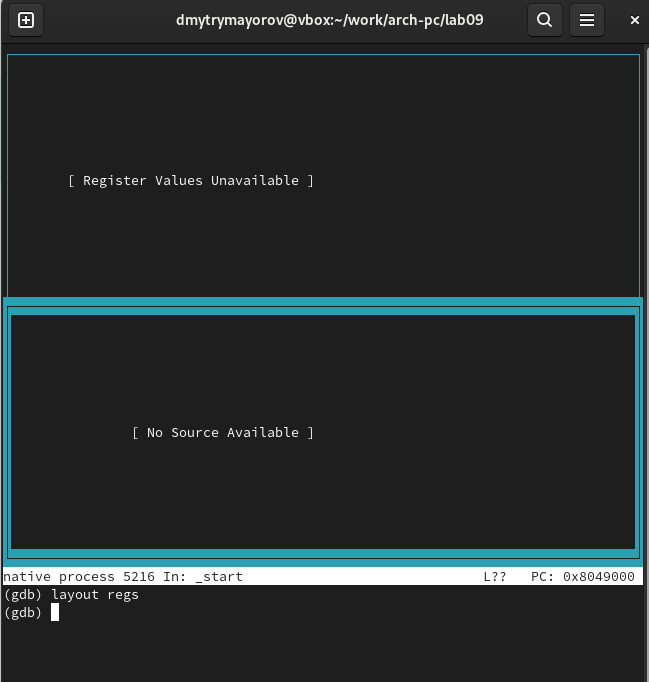


Смотрим дисассимилированный код программы

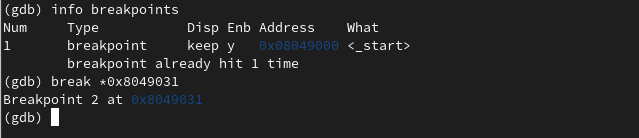


Переключаемся на синтаксис intel

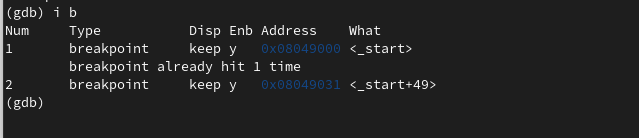
Различия отображения синтаксиса машинных команд в режимах ATT и Intel: 1.Порядок операндов: В ATT синтаксисе порядок операндов обратный, сначала указывается исходный операнд, а затем - результирующий операнд. В Intel син- таксисе порядок обычно прямой, результирующий операнд указывается первым, а исходный - вторым. 2.Разделители: В ATT синтаксисе разделители операндов - запятые. В Intel синтаксисе разделители могут быть запятые или косые черты (/). 3.Префиксы размера операндов: В ATT синтаксисе размер операнда указывается перед операндом с использованием префиксов, таких как “b” (byte), “w” (word), “l” (long) и “q” (quadword). В Intel синтаксисе размер операнда указывается после операнда с использованием суффиксов, таких как “b”, “w”, “d” и “q”. 4.Знак операндов: В ATT синтаксисе операнды с позитивными значениями предваряются символом ““.””. 5.Обозначение адресов: В ATT синтаксисе адреса указываются в круглых скобках. В Intel синтаксисе адреса указываются без скобок. 6.Обозначение регистров: В ATT синтаксисе обозначение регистра начинается с символа “%”. В Intel синтаксисе обозначение регистра может начинаться с символа “R” или “E” (например, “%eax” или “RAX”).



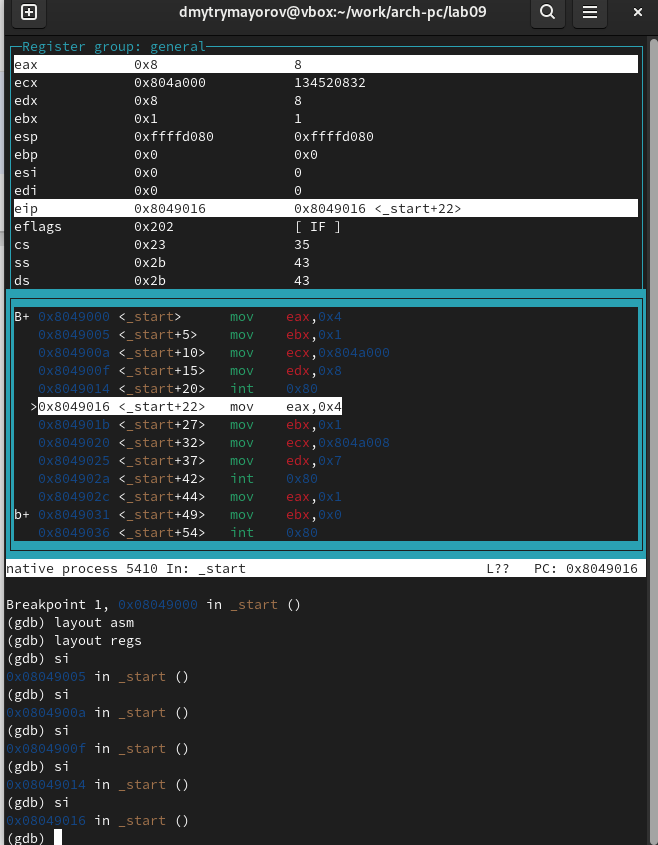
Включаем отображение регистров, их значений и результат дисассимилирования программы



Создаем новую точку останова



Смотрим информацию



Отслеживаем регистры командой si

Во время выполнения команд менялись регистры: ebx, ecx, edx,eax, eip.

Смотрим значение переменной msg1 по имени

Смотрим значение переменной msg1 по имени

Смотрим значение переменной msg2 по адресу

Смотрим значение переменной msg2 по адресу



Изменим перввй символ переменной msg1



Изменим перввй символ переменной msg2

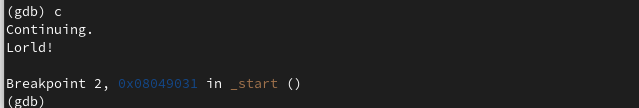


Смотрим значение регистра edx в разных форматах



Изменяем регистр ebx

Выводится разные значения, так как команда без кеавычек присваивает регистру вводимое значение.



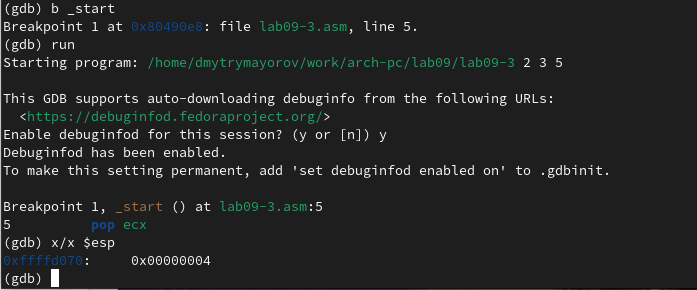
Прописываем команды для завершения программы и выхода из GDB

Копируем файл lab8-2.asm в файл с именем lab09-3.asm

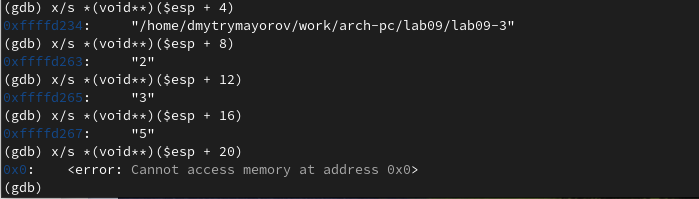
Копируем файл lab8-2.asm в файл с именем lab09-3.asm

Создаем исполняемый файл и запускаем его в отладчике GDB

Создаем исполняемый файл и запускаем его в отладчике GDB



Устанавливаем точку останова



Изучаем полученные данные

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

# 3 Задание для самостоятельной работы

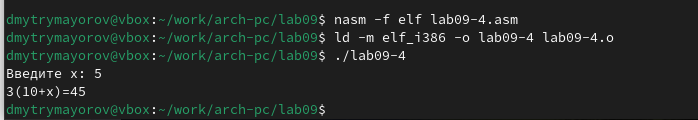
Задание 1

Копируем файл lab8-4.asm в файл с именем lab09-4.asm

Копируем файл lab8-4.asm в файл с именем lab09-4.asm



Изменяем файл. СОздаем подпрограмму

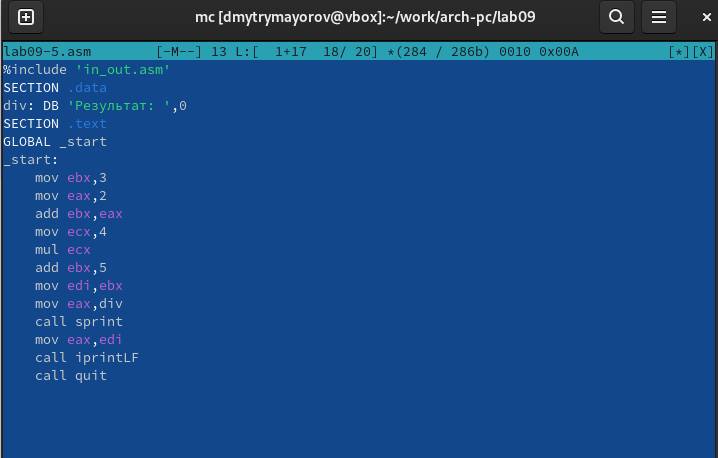


Создаем исполняемый файл и запускаем его

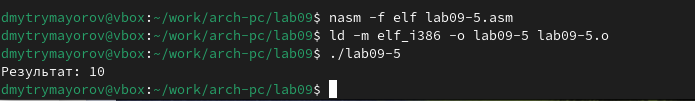
Задание 2

Создаем файл

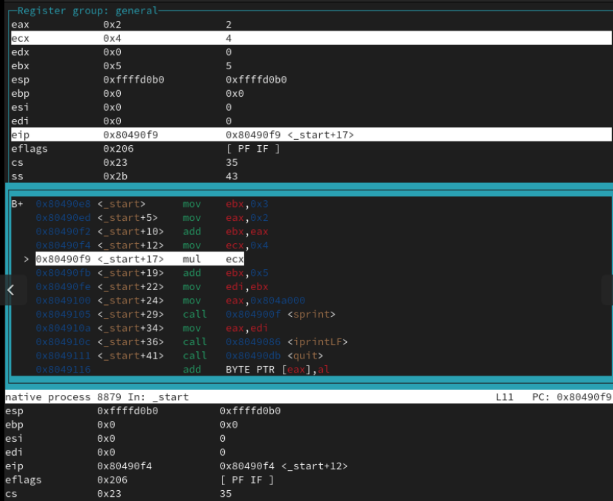
Создаем файл



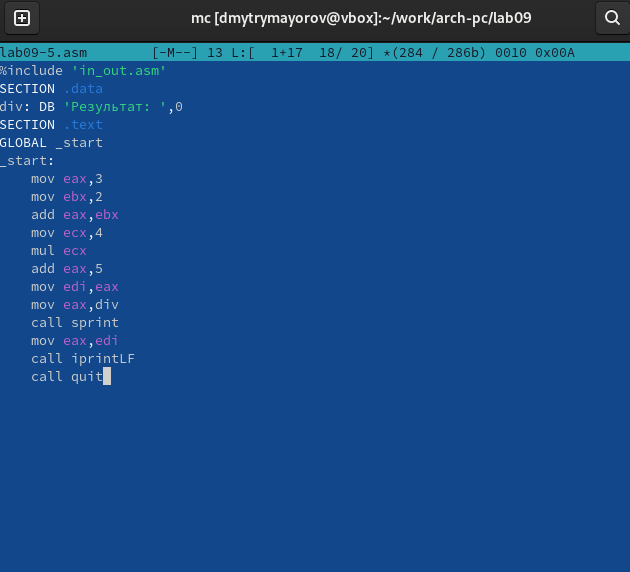
ОТкрываем файл и заполняем его в соответсвии с листингом



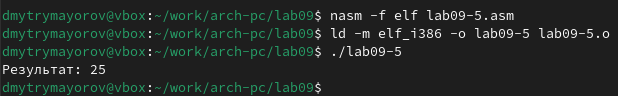
Создаем исполняемый файл и запускаем его. Программа работает неправильно



Ищем ошибку регистров в отладчике



Изменяем файл



Создаем исполняемый файл и запускаем его. Программа работает правильно

# 4 Выводы

Мы познакомились с методами отладки при помощиGDBи его возможностями.

# Список литературы