Отчёт по лабораторной работе №8

Дисциплина: Комьютерные технологии и технологии программирования

Дымовой Д.Д.

Содержание

# 1 Цель работы

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

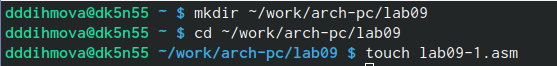
# 2 Теоретическое введение

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 пре- рывание выполнения программы (например, это ошибки, связанные с переполнением или делением на ноль).

Наиболее часто применяют следующие методы отладки: • создание точек контроля значений на входе и выходе участка программы (например, вывод промежуточных значений на экран — так называемые диагностические сообще- ния); • использование специальных программ-отладчиков.

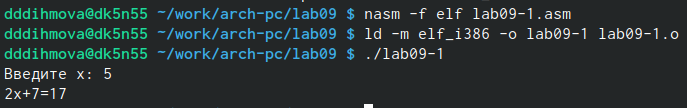
# 3 Выполнение лабораторной работы

Создаю каталог для выполнения лабораторной работы №9, перехожу в него, создаю ассемблеровский файл для работы (рис. ??).



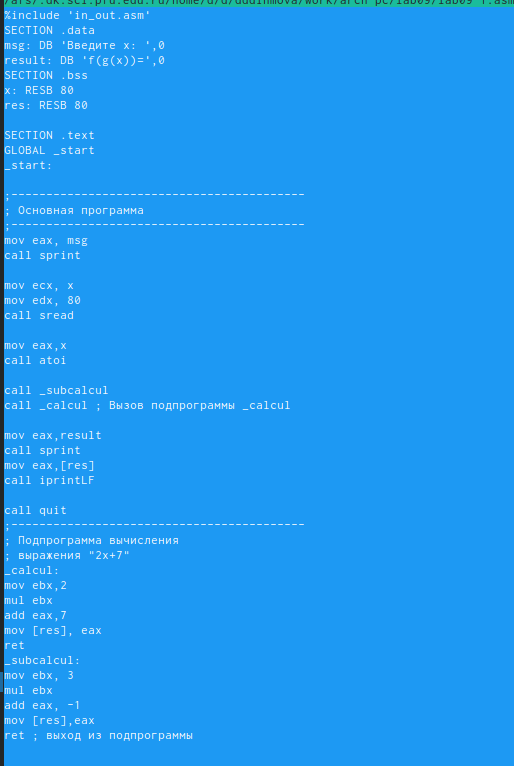
Cоздание каталога

Копирую текст листинга 9.1, создаю исполняемый файл и запускаю его (рис. ??).



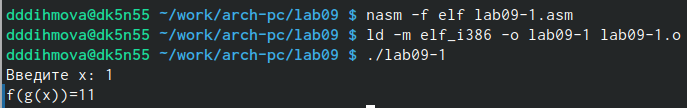
Запуск исполняемого файла

Изменяю текст программы листинга 9.1, чтобы программа работала так, как указано в условии лабораторной работы (рис. ??).



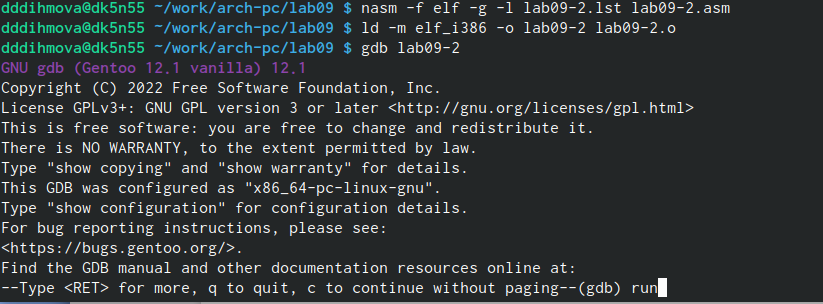
Программа

Создаю исполняемый файл и запускаю его (рис. ??).



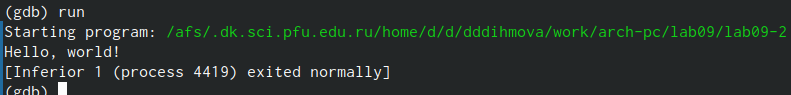
Программа f(g(x))

Создаю файл lab09-2.asm, вставляю туда текст программы листинга 9.2, сохраняю, создаю исполняемый файл и запускаю его (рис. ??).



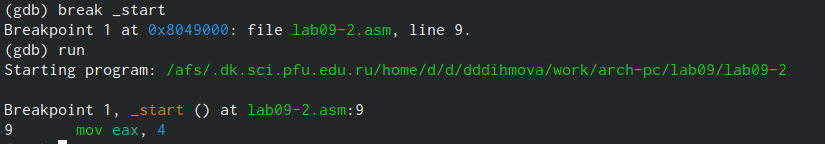
Запуск файла

Запускаю файл в оболочке GDB (рис. ??).



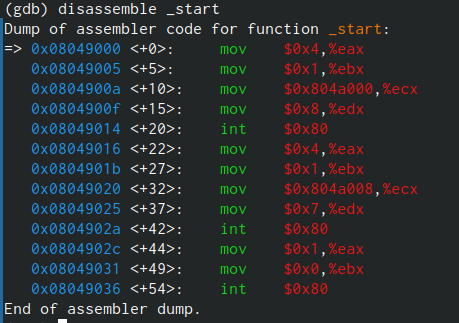
Оболочка GDB

Устанавливаю брейкпоинт на метку \_start (рис. ??).



Установка брейкпоинта

Просматриваю дисассимилированный код программы начиная с метки (рис. ??).



Просмотр кода программы

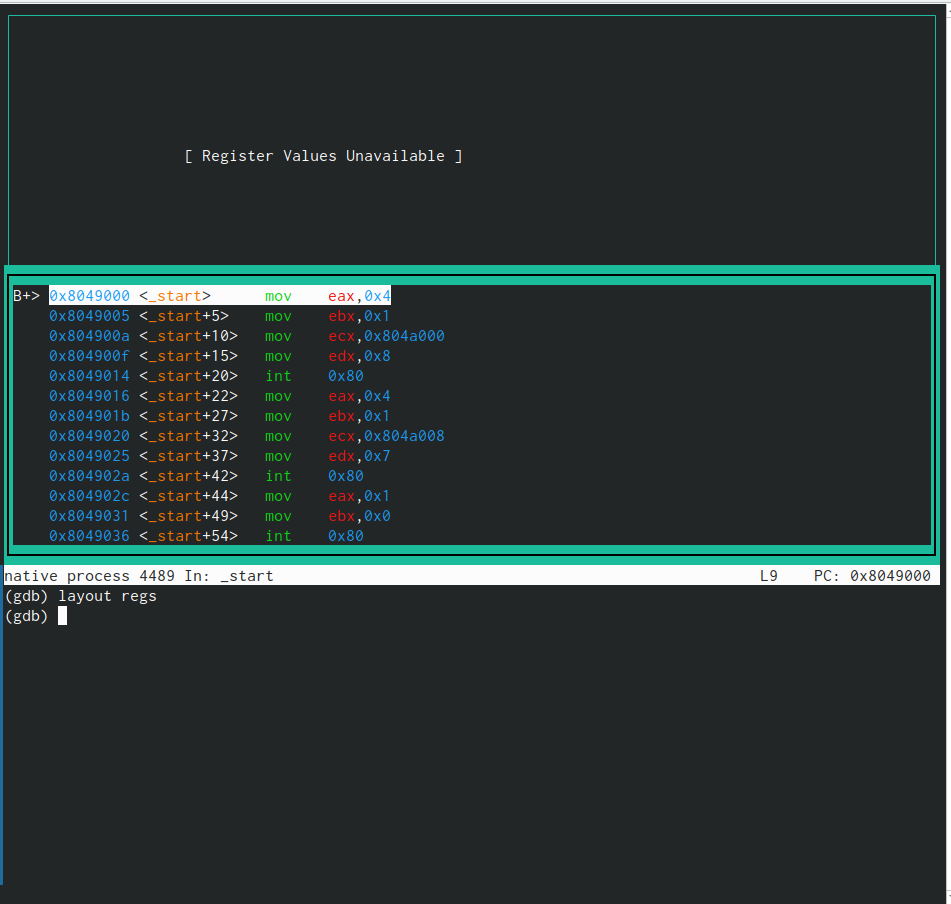
Переключаюсь на отображение команд с Intel’овским синтаксисом, введя команду set disassembly-flavor intel (рис. ??).



Переключение

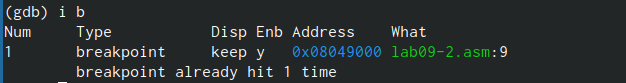
ПЕРЕЧИСЛИТЬ РАЗЛИЧИЯ ОТОБРАЖЕНИЯ СИНТАКСИСА

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



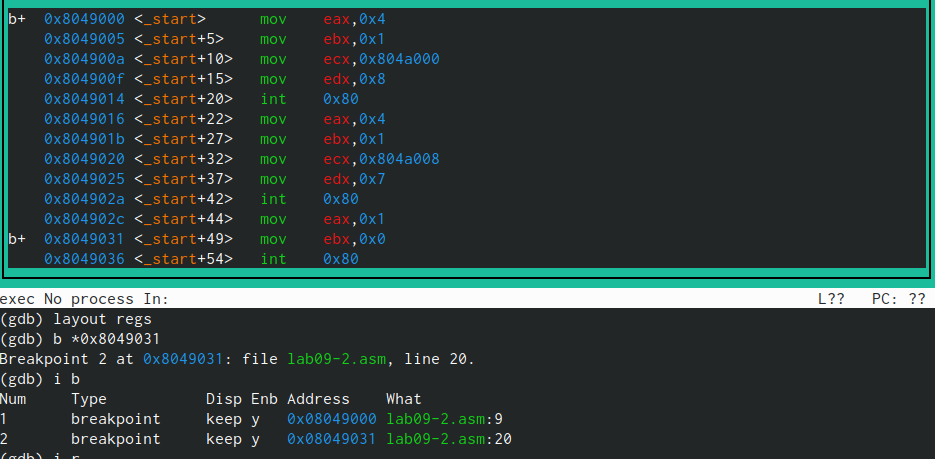
Переключение режима

Просмотр точек брейкпоинта (рис. ??).



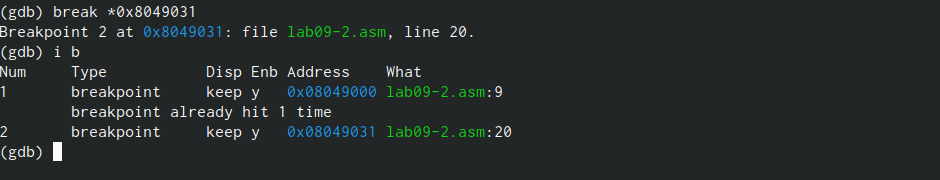
Точки брейкпоинта

Устанавливаю брейкпоинт на инструкцию mov ebx, 0x0 (рис. ??).



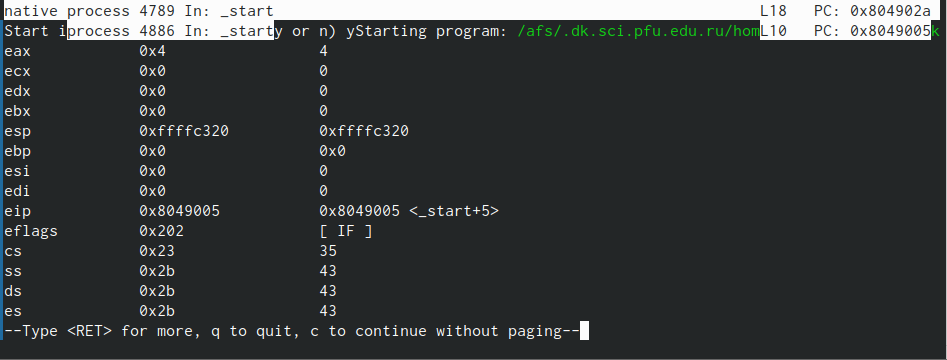
Брейкпоинт

Информация о всех точках останова (рис. ??).



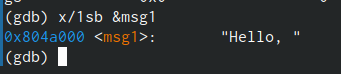
Точки останова

Просматриваю содержимое регистров с помощью команды i r (рис. ??).



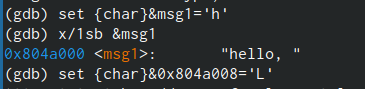
Информация о содержимом регистров

Смотрю значение переменной msg1 (рис. ??).



Значение переменной

Заменяю значение переменной msg1 (рис. ??).



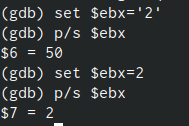
Изменение переменной msg1

Просматриваю значение аргумента msg2. 0x804a008 адрес msg2 (рис. ??).

Значение аргумента msg2

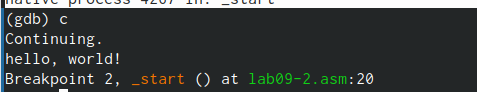
Значение аргумента msg2

Просматриваю значение регистра ebx в разных форматах (рис. ??).



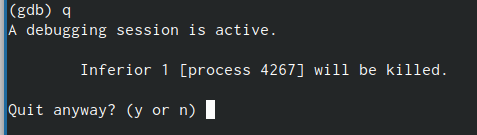
Значение регистра ebx

Завершаю выполнение программы с помощью команды q (рис. ??).



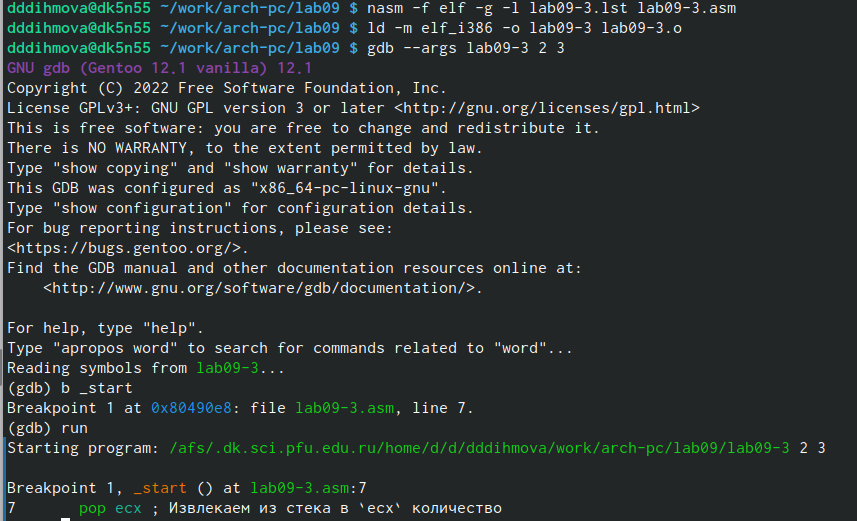
Завершение программы

Выхожу из gdb с помощью команды q (рис. ??).



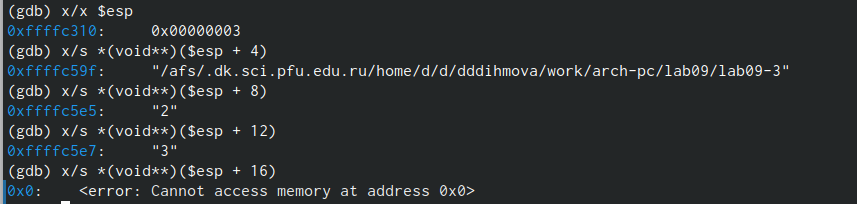
Выход из откладки

Я копирую файл lab8-2.asm в папку с лабораторной номер 9 и называю его lab09-2.asm, создаю исполняемый файл и запускаю его через gdb. Эта программы должна находить произведение аргументов. Также я ставлю точку останова на месту старт (рис. ??).



Запуск программы с аргументами 2 3, установка брейкпоинта

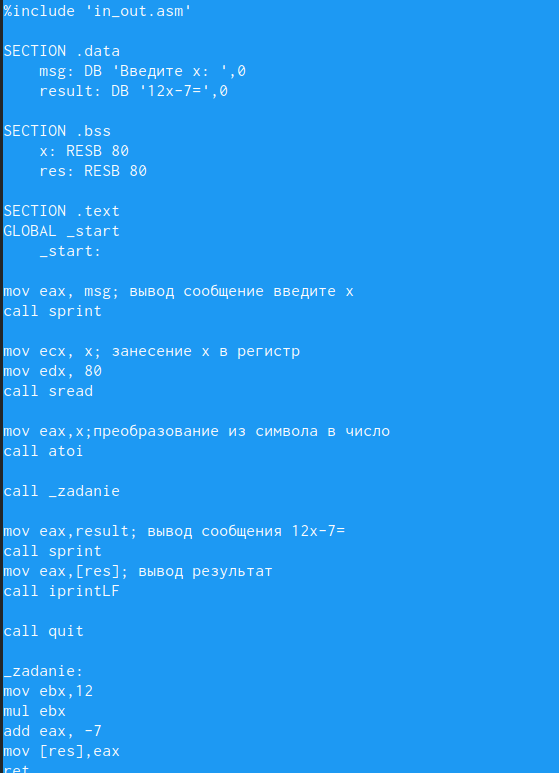
Рассматриваю позиции стека, так как у меня всего два аргумента на шаге ( +16) выдаёт ошибку (рис. ??).



Позиции стека

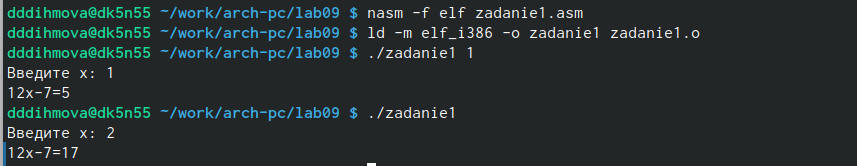
# 4 Задание для самостоятельной работы

Создаю файл zadanie1.asm для выполнения первого задания из самостоятельной работы. Ввожу текст программы листинга 9.1 для удобства. Пишу программу согласно 13 варианту лабораторной работы номер 8 (рис. ??).



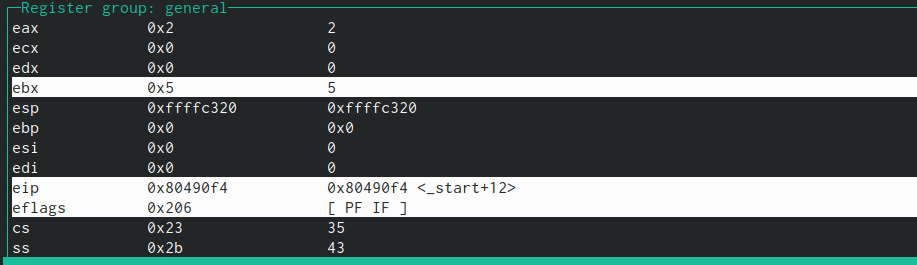
Мною преобразованная программа

Создаю исполняемый файл и запускаю его (рис. ??).



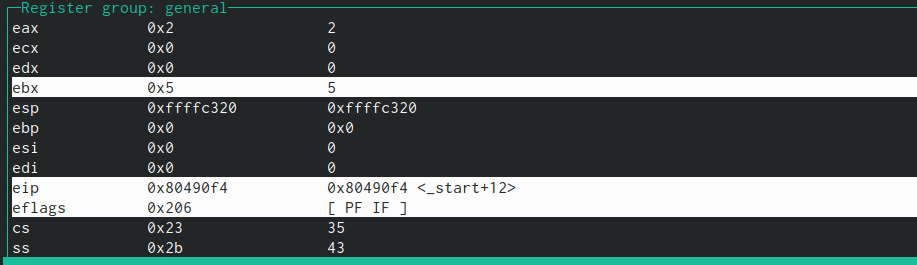
Запуск файла

Создаю файл zadanie2.asm для выполнения второго пункта лабораторной работы, ввожу туда текст листинга 9.3, сохраняю, создаю исполняемый файл и запускаю его. Убеждаюсь, что результат неверный. Ответ должен быть 25 (рис. ??).



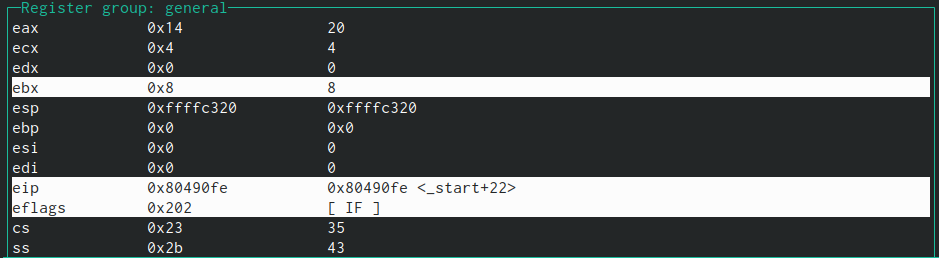
Запуск файла

Запускаю откладчик, смотрю как изменяется регистры eax, ebx, ecx пошагово с помощью команды si. Замечаю, что на третьем шаге регистр ebx имеет значение 5, а на четвертом шаге ecx и eax перемножаются, что и дает неверный результат. Следовательно изменяю программу так, чтобы результат сложения записывался в eax (рис. ??).



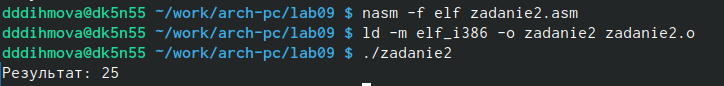
Регистр ebx=5, а должно быть eax=5

У меня снова вывелся неправильный результат, пошагово проверяю как изменяются регистры, замечаю что в следующем сложении 5 прибавляется к регистру ebx, а должно к регистру eax, и в конце результат edi записывается в ebx, а должен в eax, изменяю и эту строчку (рис. ??).



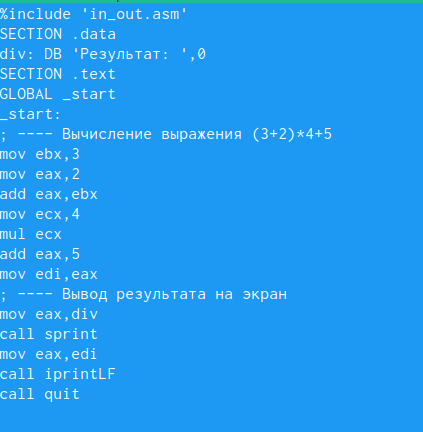
Ошибка в сложении

Создаю исполняемый файл и запускаю его (рис. ??).



Проверка программы

Текст моей программы (рис. ??).



Программа

# 5 Выводы

Я приобрела навыки написания программ с использованием подпрограмм. Познакомилась с методами отладки при помощи GDB и его основными возможностями.

# Список литературы