Лабораторная работа №10

Дисциплина: Архитектура компьютера

Кондратьев Арсений Вячеславович

07.10.2022

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

1. Реализовал пример программы с использованием вызова подпрограммы(рис.1)

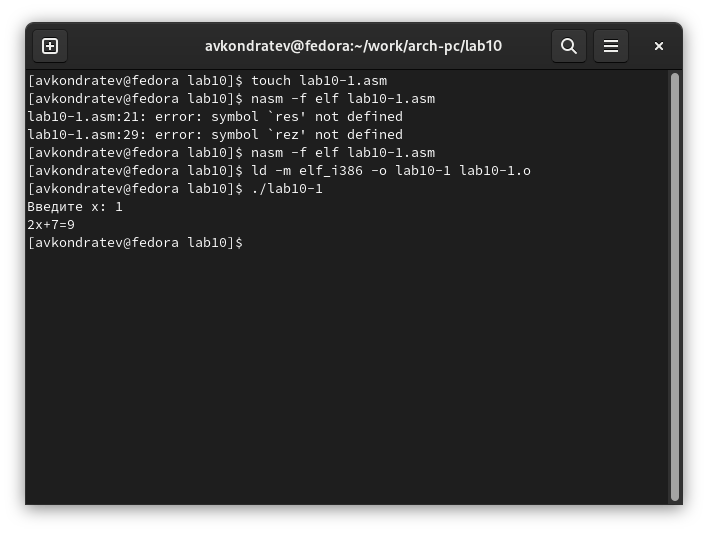


Figure 1: Рис. 1

1. Реализовал программу вывода сообщения Hello world! и проверил работу программы, запустив ее в оболочке GDB с помощью команды run (рис.2)

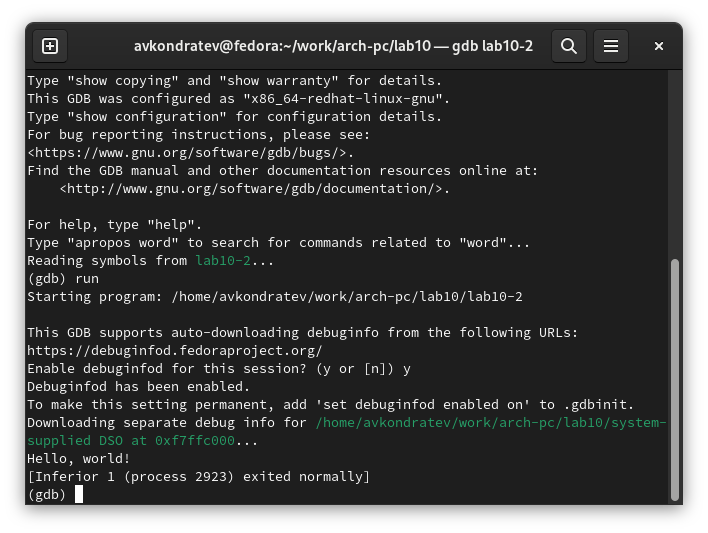


Figure 2: Рис. 2

1. Установил брейкпоинт на метку \_start (рис.3)

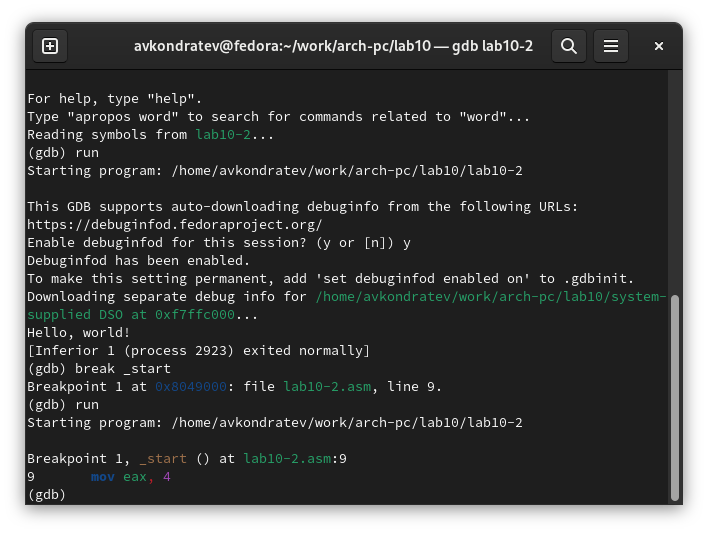


Figure 3: Рис. 3

1. Посмотрел дисассимилированный код программы(рис.4)

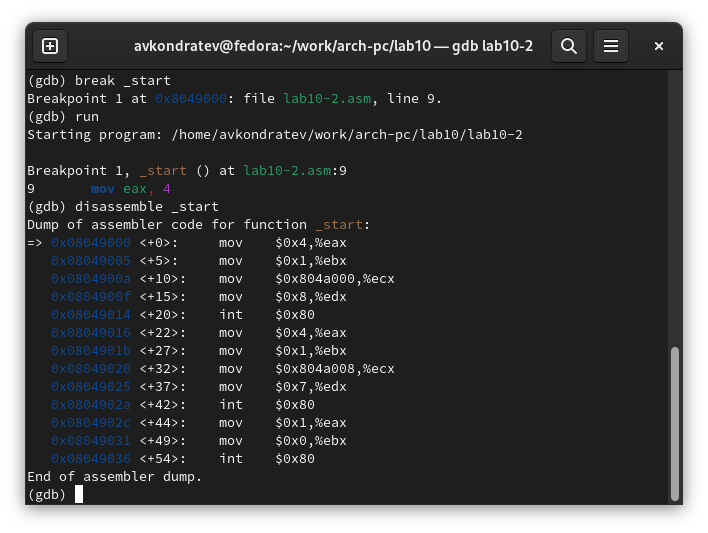


Figure 4: Рис. 4

1. Переключился на отображение команд с Intel’овским синтаксисом(рис.5)

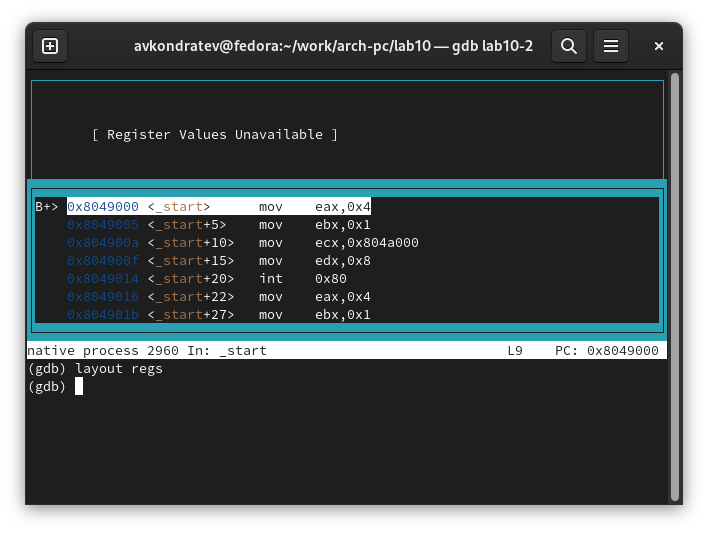


Figure 5: Рис. 5

1. Посмотрел значение переменной msg2 по адресу и изменил значение ячейки памяти(рис.6)

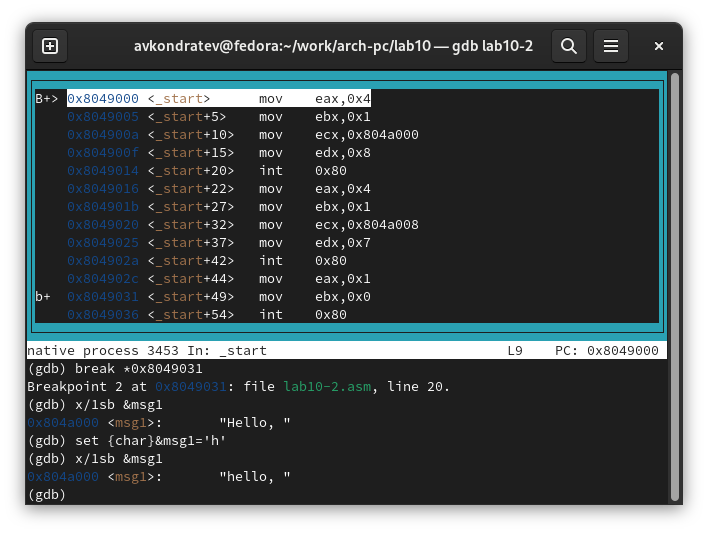


Figure 6: Рис. 6

1. Заменил символ во второй переменной msg2(рис.7)

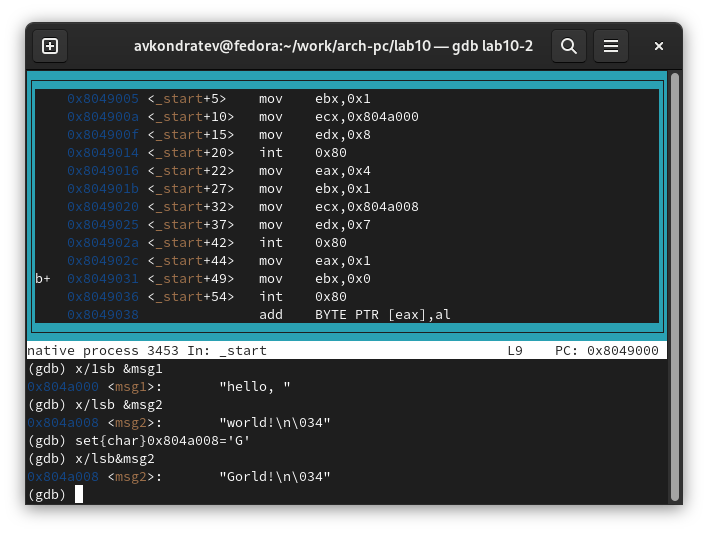


Figure 7: Рис. 7

1. Посмотрел количество аргументов командной строки(рис.8)

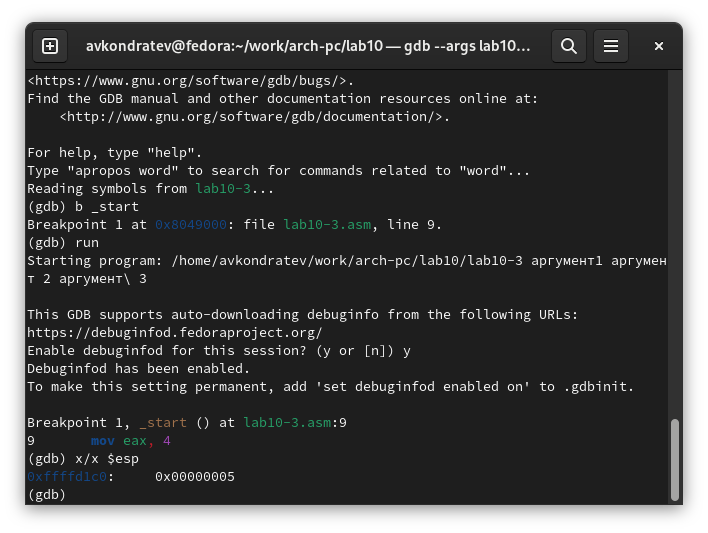


Figure 8: Рис. 8

1. Преобразовал программу из лабораторной работы №9, реализовав вычисление значения функции как подпрограмму(рис.9)(рис.10)

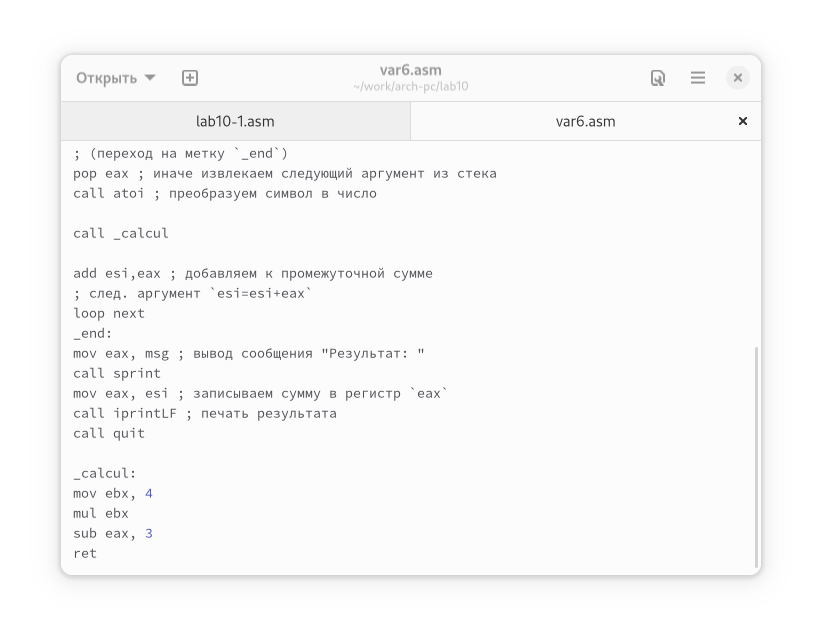


Figure 9: Рис. 9

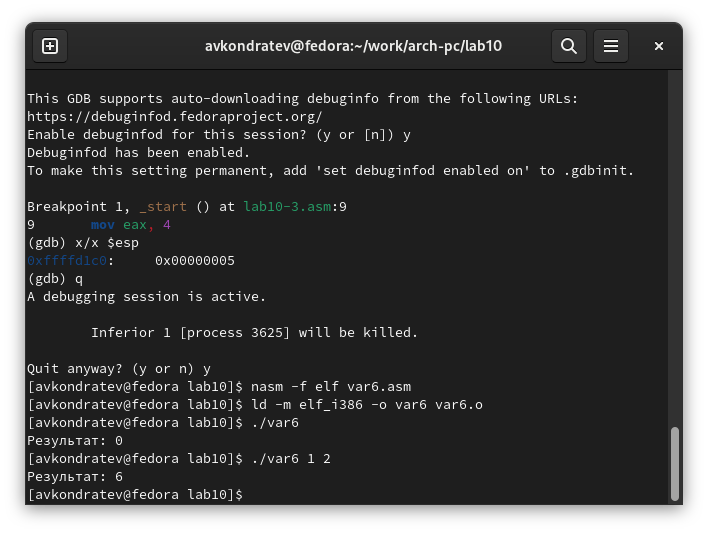


Figure 10: Рис. 10

1. С помощью отладчика GDB, анализируя изменения значений регистров, определил ошибку и исправил ее(рис.11)(рис.12)(рис.13)

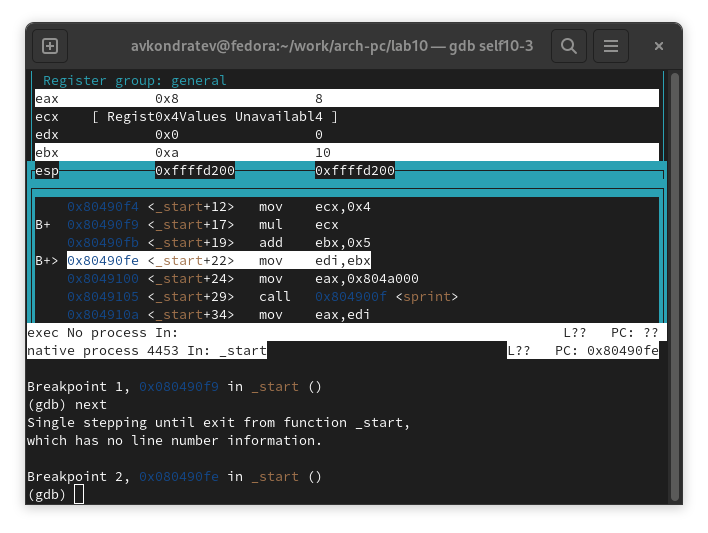


Figure 11: Рис. 11

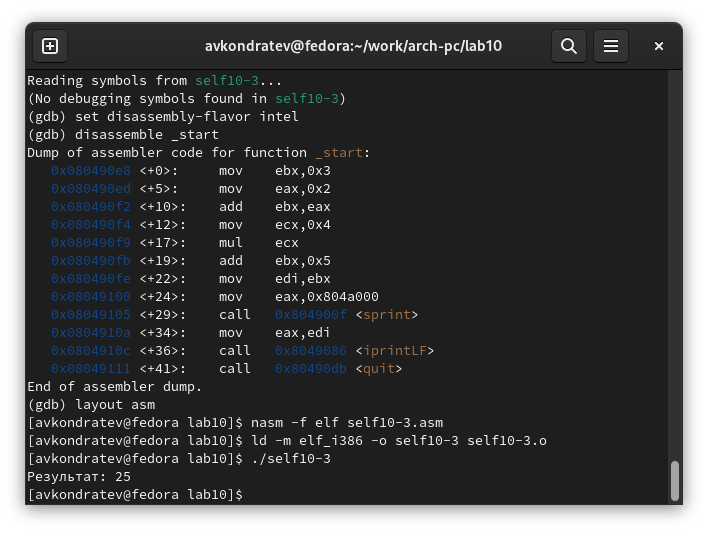


Figure 12: Рис. 12

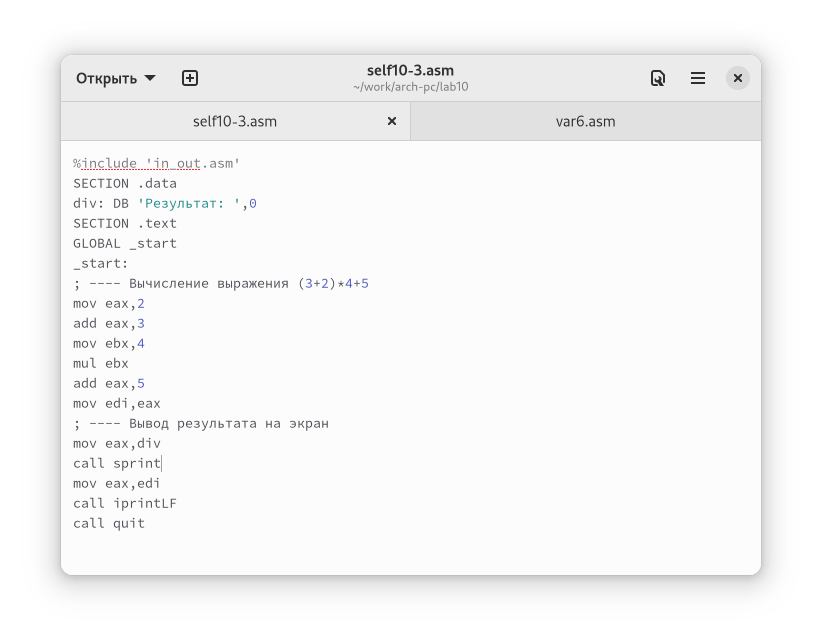


Figure 13: Рис. 13

# 3 Выводы

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

# 4 Контрольные вопросы

1. Инструкция call и инструкция ret
2. С помощью инструкции call и имени подпрограммы
3. Для вызова подпрограммы из основной программы используется инструкция call, которая заносит адрес следующей инструкции в стек и загружает в регистр eip адрес соответствующей подпрограммы. Подпрограмма завершается инструкцией ret, которая извлекает из стека адрес, занесённый туда соответствующей инструкцией call, и заносит его в eip.
4. Операнд необязателен, но если он есть, то после того, как будет считан адрес возврата, из стека будет удалено столько байтов, сколько указано в операнде
5. Отладчики позволяют управлять ходом выполнения программы, контролировать и изменять данные. Это помогает быстрее найти место ошибки в программе и ускорить её исправление
6. В отладке можно видеть текущие(во время останова) значения регистров, их адреса. С помощью GDB run
7. breakpoint - это преднамеренное прерывание выполнения программы, при котором выполняется вызов отладчика  
   сheckpoint - снимок состояния программы  
   Watchpoint - точка останова по данным. Срабатывает, когда меняется значение заданного выражения или переменной  
   Catchpoint - специальная точка останова, которая срабатывает при достижении специального события, например, C++ исключения или загрузки библиотеки  
   Call stack (стек вызовов) - хранит информацию об активных процедурах и функциях
8. disassemble - Посмотреть дизассемблированный код программы  
   run - запуск в режиме отладки  
   breakpoint - установить точку останова  
   kill (сокращённо k) прекращает отладку программы  
   next - переход к следующей точке останова