Отчёт по лабораторной работе №9

Ярослав Антонович Меркулов

Содержание

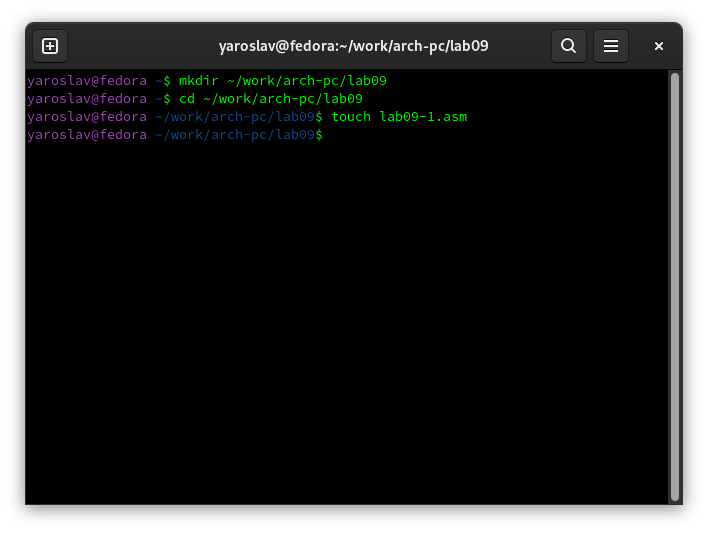
# 1 Цель работы

Приобрести навыки написания программ с использованием подпрограмм. Ознакомиться с методами отладки при помощи GDB.

# 2 Выполнение лабораторной работы

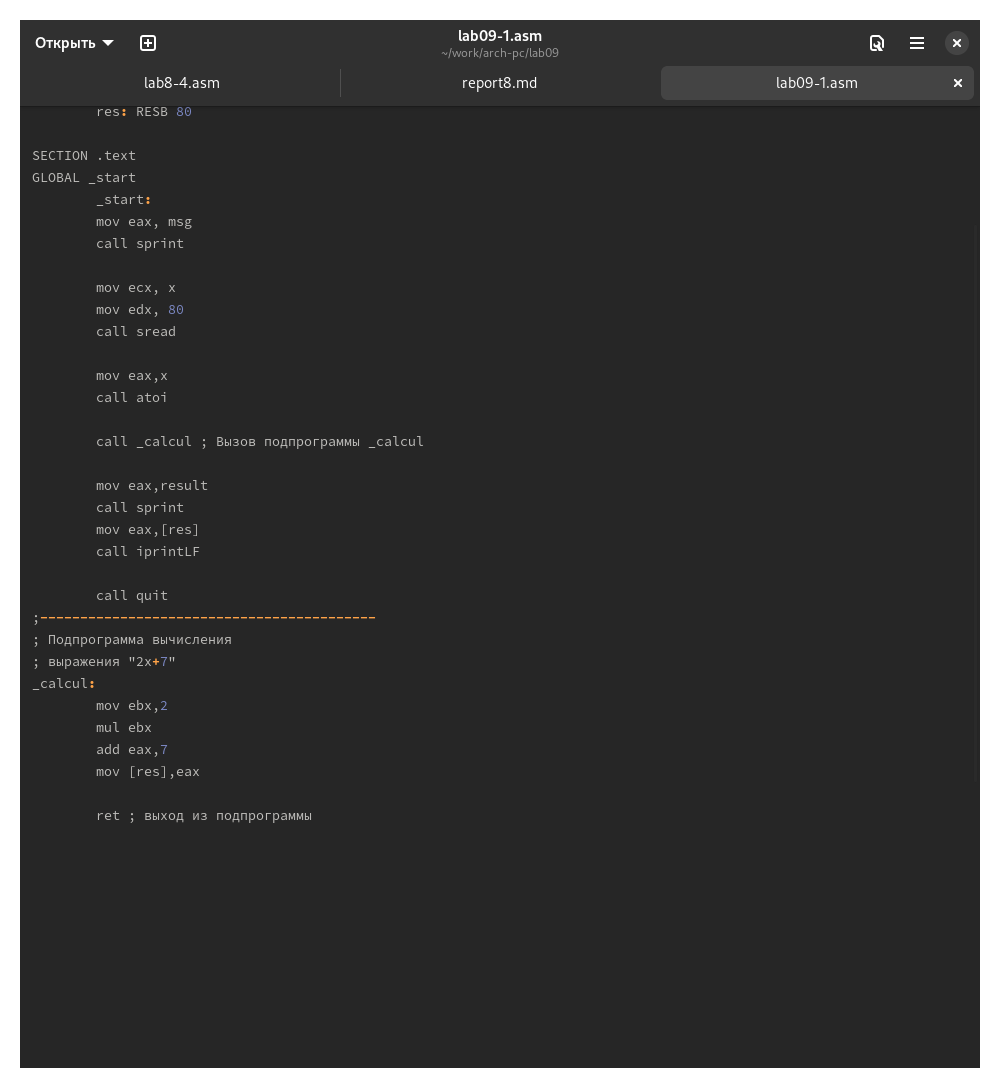
## 2.1 Реализация подпрограмм в NASM

1. Создаём каталог для лабораторной работы, переходим в него и создаём файл lab09-1.asm.



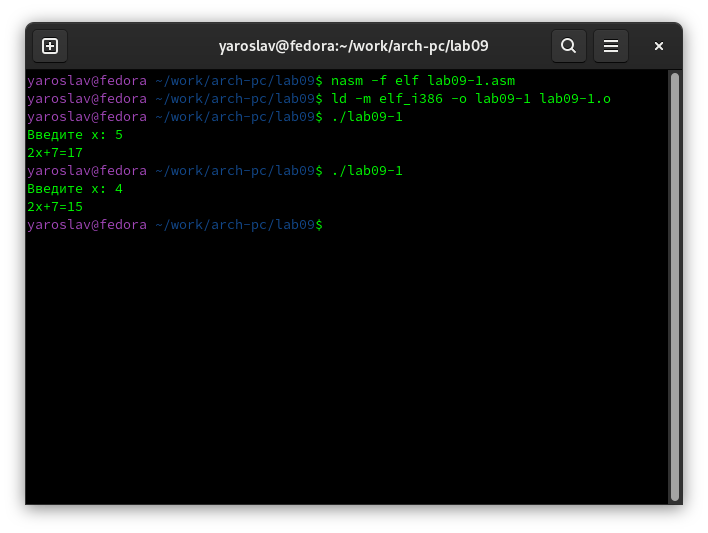
Создание каталога, переход в него, создание файла

1. Вводим текст программы из листинга 9.1.



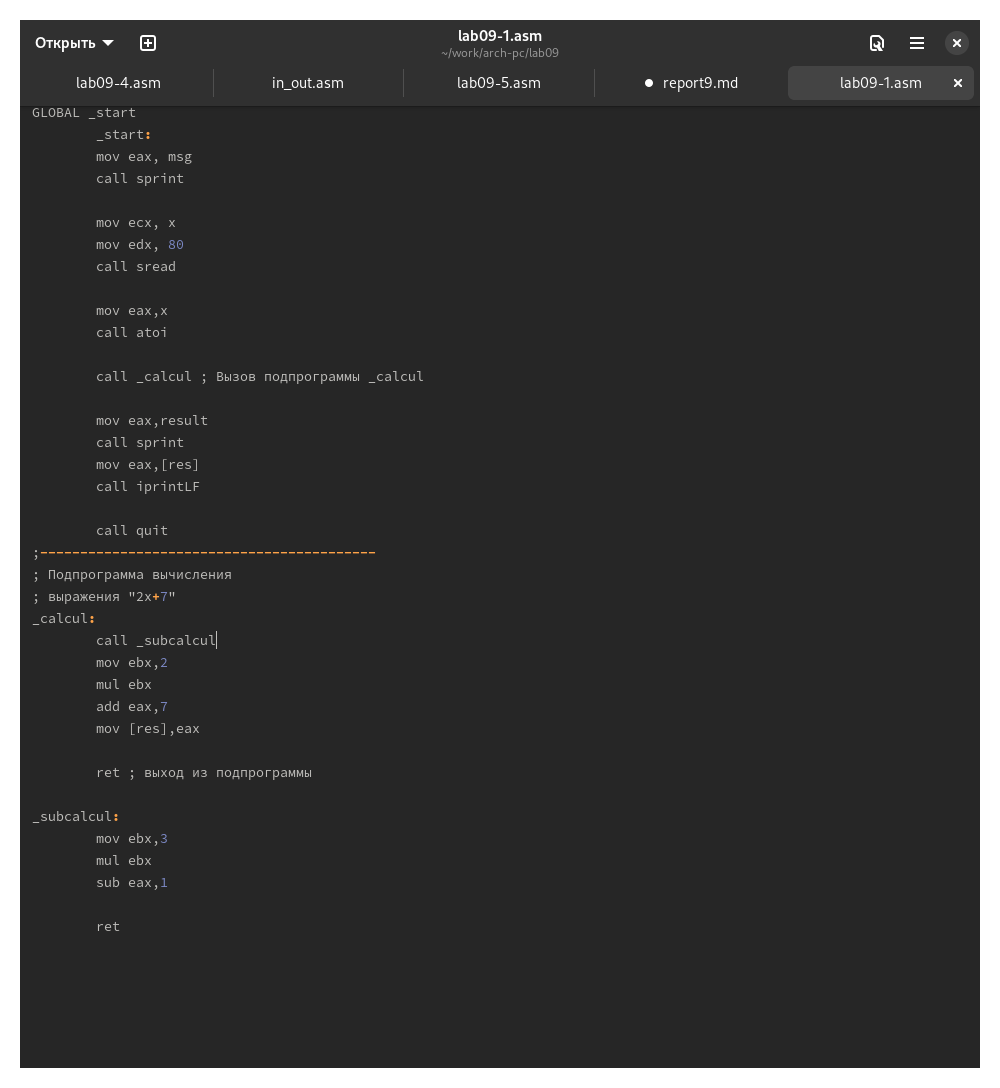
Введённый текст программы

1. Создаём исполняемый файл и тестируем.

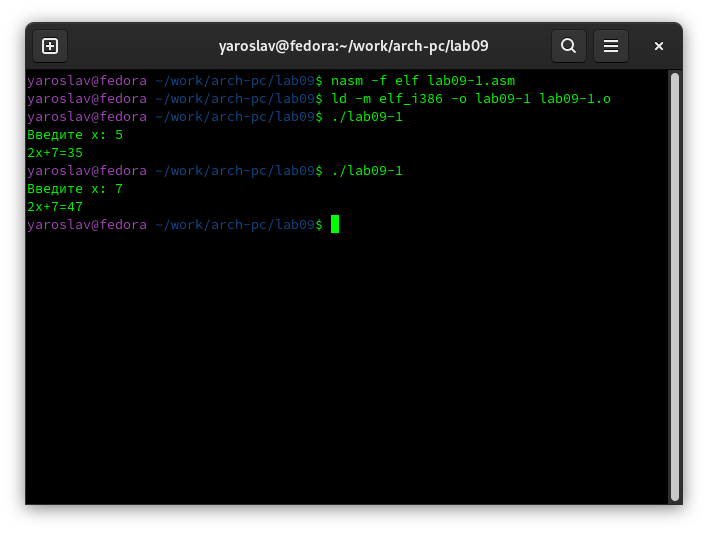


Работа программы

1. Изменяем текст программы, создаём подпрограмму \_subcalcul.



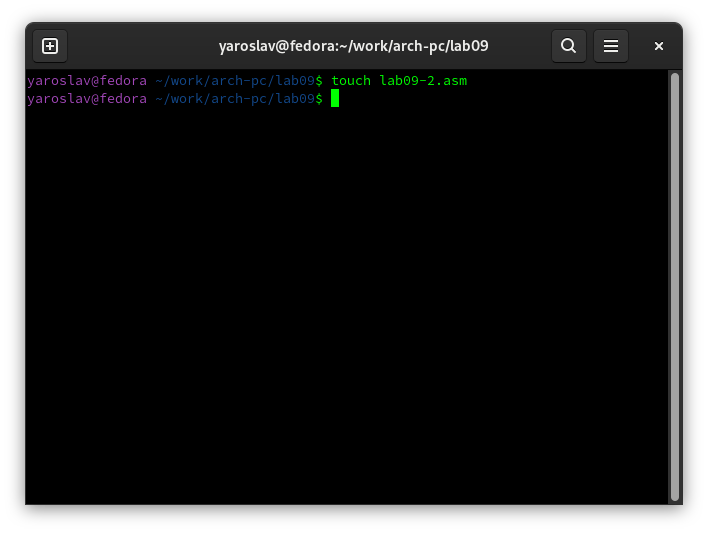
Изменённый файл



Работа новой программы

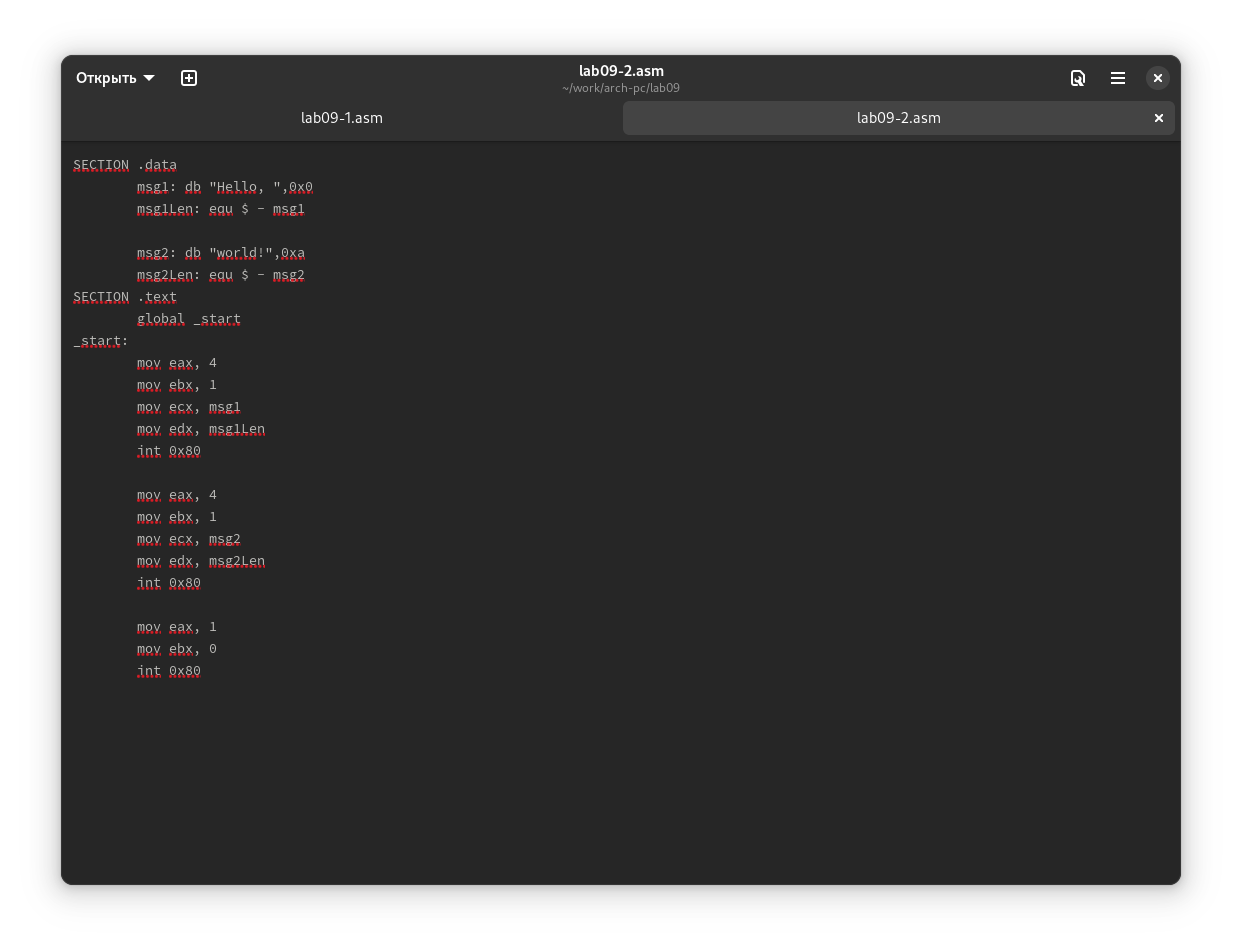
## 2.2 Отладка программам с помощью GDB

1. Создаём файл lab09-2.asm.



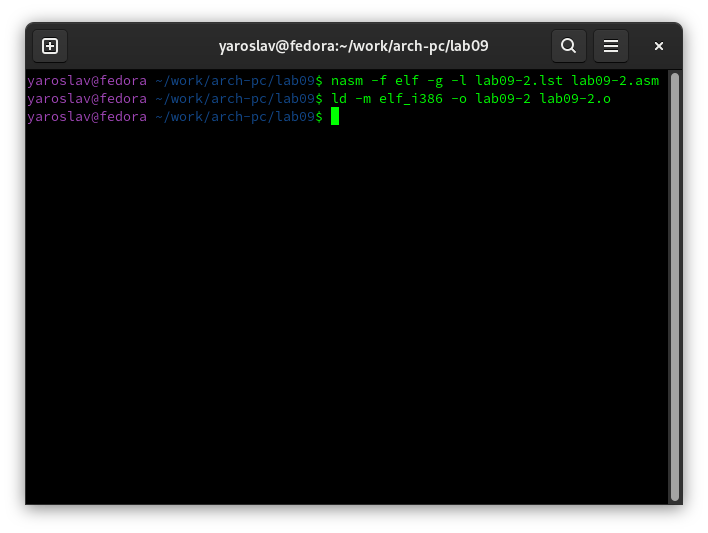
Создание файла

1. Записываем в файл листинг.



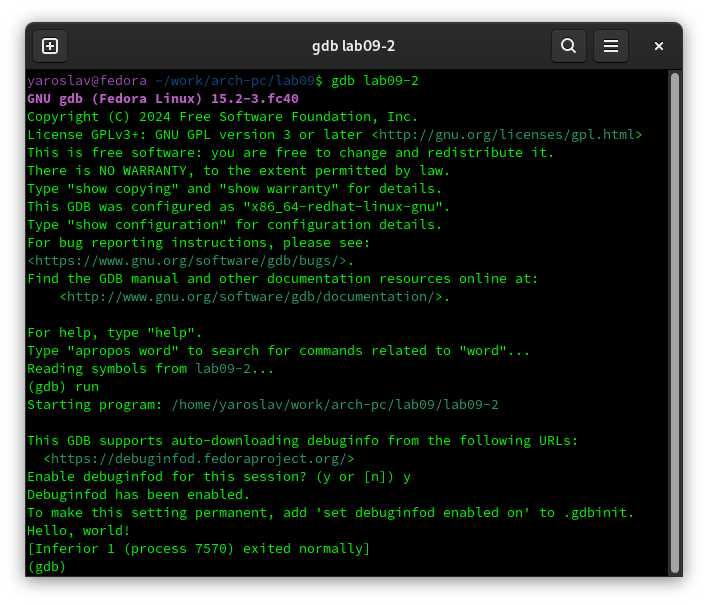
Lab09-2.asm

1. Получаем исполняемый файл.



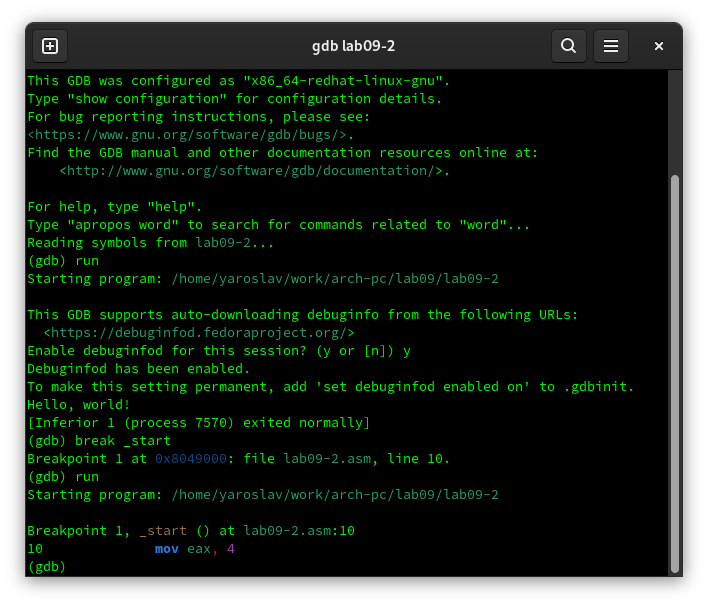
Компиляция и трансляция с параметрами

1. Запускаем отладчик, запускаем в нём программу.



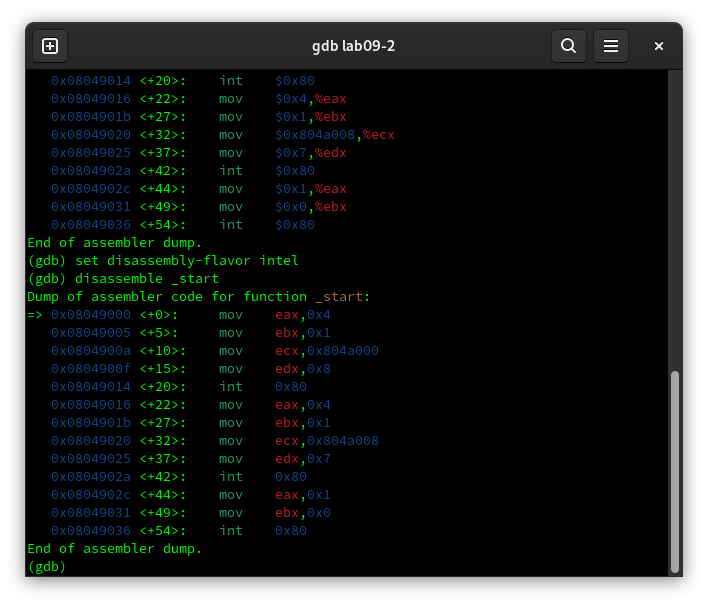
Отладчик

1. Ставим breakpoint на \_start.



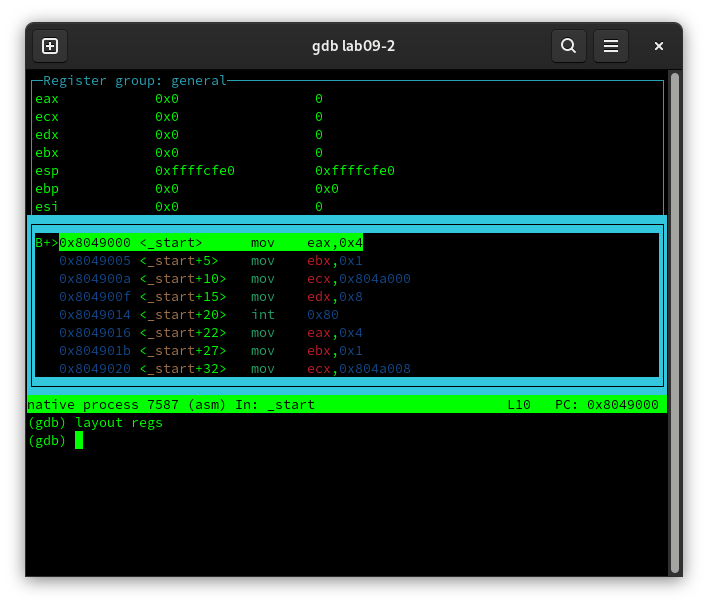
Брейкпоинт

1. Смотрим дисассимилированный код программы, переключаем отображение.



Дисассимилированный код

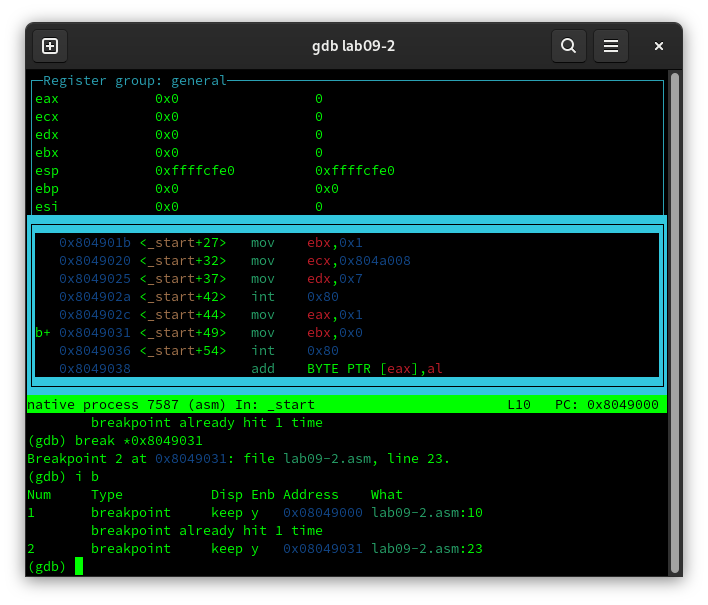
1. Включаем режим псевдографики.



Режим псевдографики

### 2.2.1 Добавление точек остановки

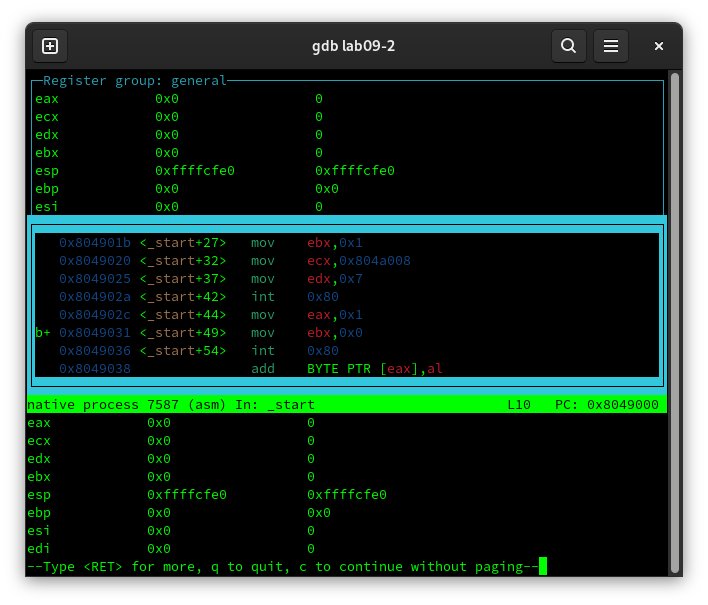
1. Ставим точку остановки и выводим о них. Видим 2 точки: первая, которую мы поставили в пункте 9, и новую.



Точки остановки

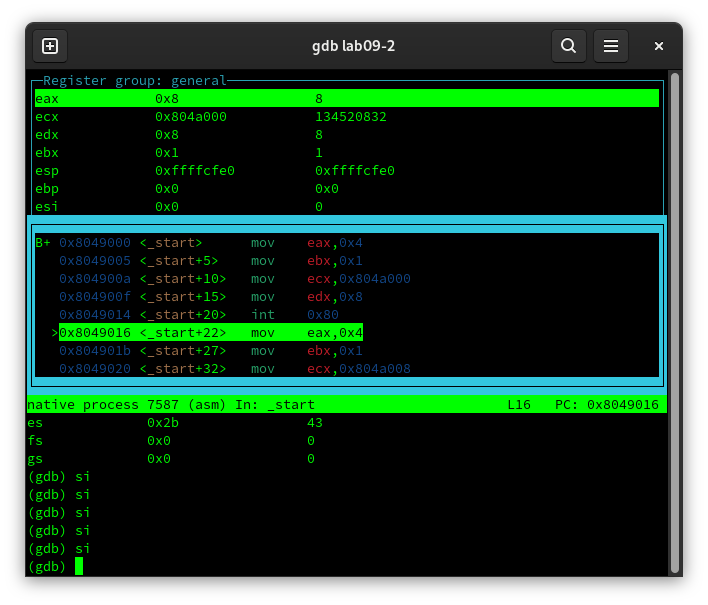
### 2.2.2 Работа с данными программы в GDB

1. Просматриваем содержимое регистров.



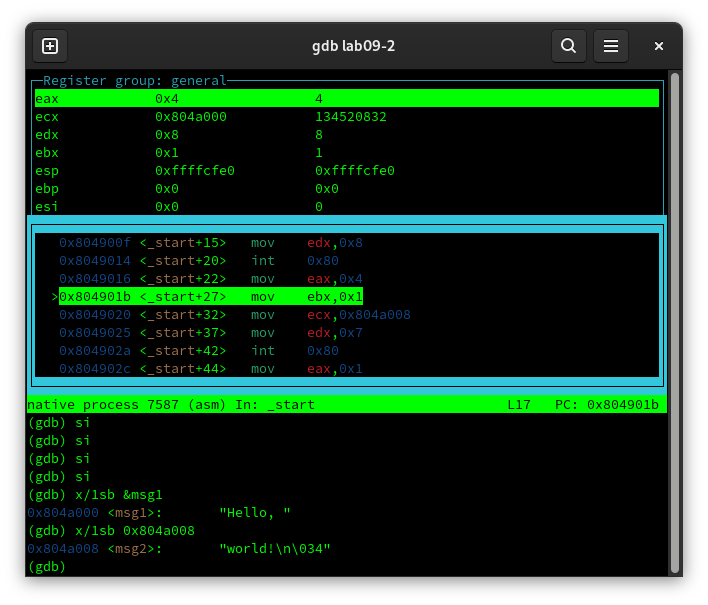
Содержимое регистров

1. Выполняем инструкции stepi (si).



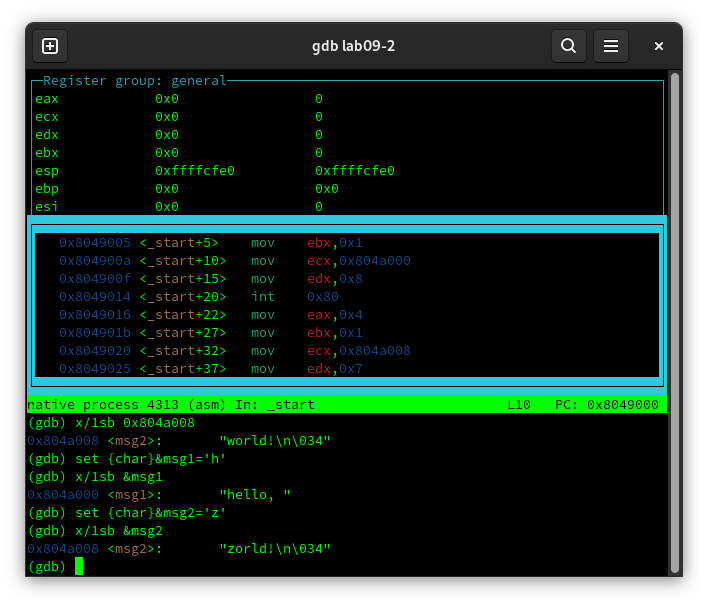
Инструкции si

1. Смотрим значение переменной по имени и адресу.



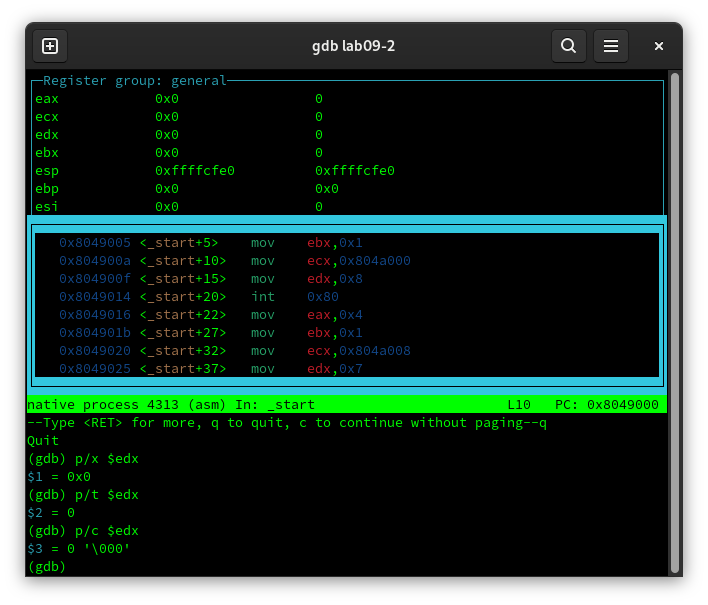
Просмотр значений переменных

1. Изменяем значения у переменных (букву).



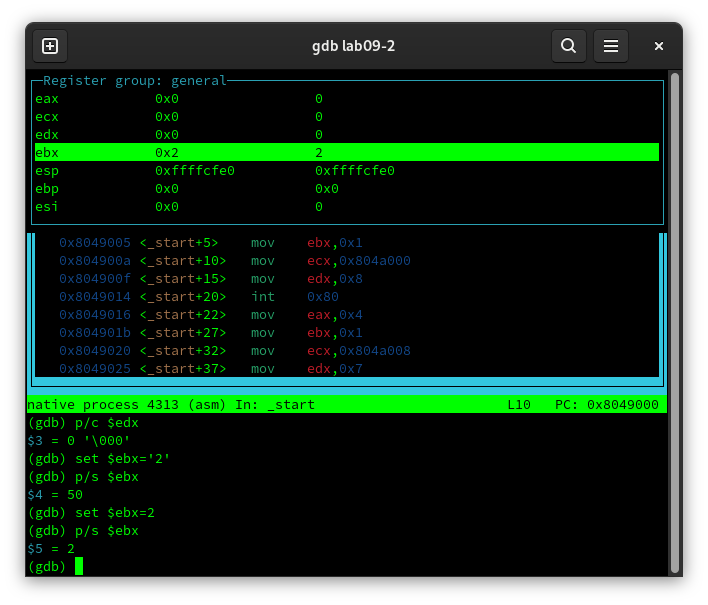
Изменение значений

1. Просматриваем регистр edx в hex,bin,char.



Значения регистра в разных регистрах

1. Меняем значения регистра ebx. В первый раз мы ввели вернулся номер по таблице ASCII у символа ‘2’. Во второй раз само число.



Изменение значений регистра

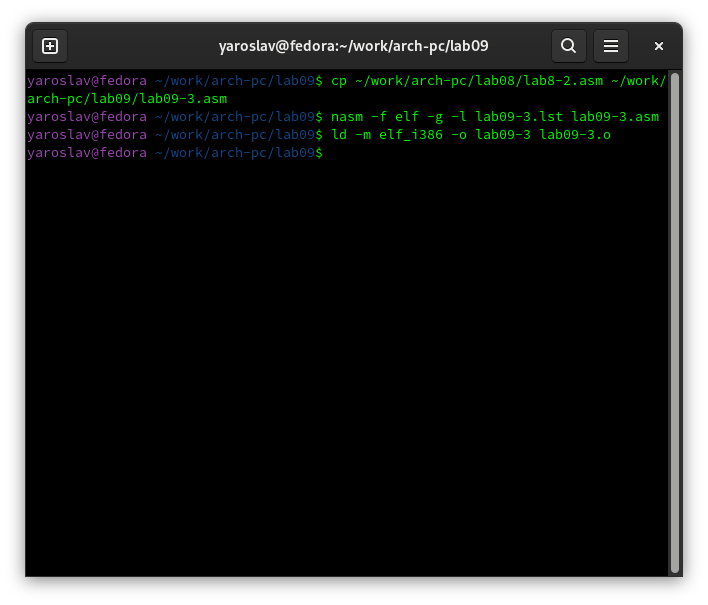
1. Завершаем выполнение программы и выходим через quit.



Завершение программы

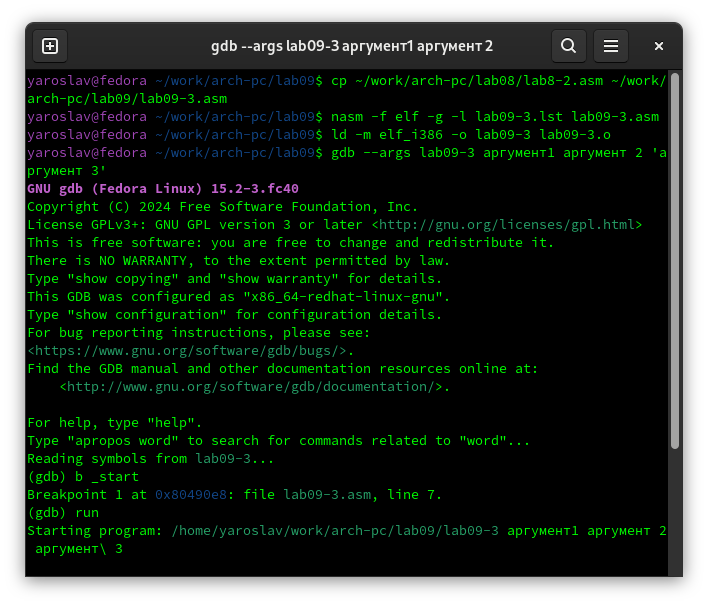
### 2.2.3 Обработка аргументов командной строки в GDB

1. Копируем файл с предыдущей лабораторной работы. Создаём исполняемый файл.



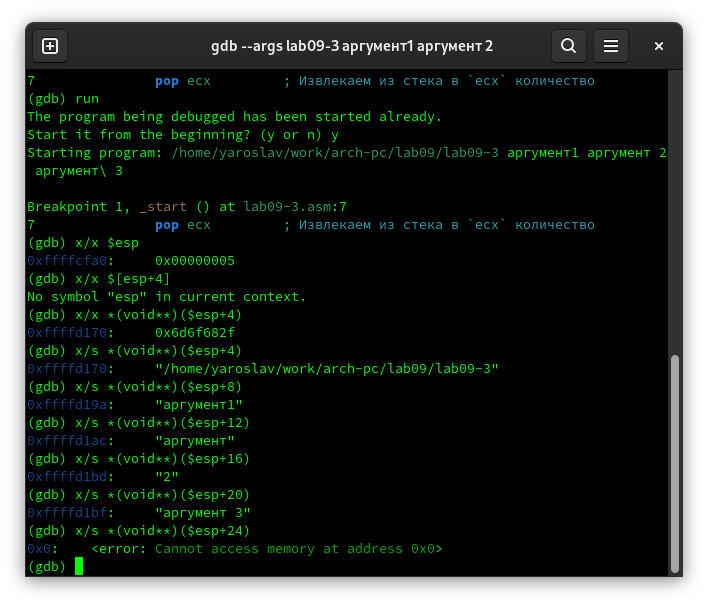
Копирование файла

1. Загружаем программу с аргументами в gdb, ставим точку остановки и запускаем.



Запуск через gdb

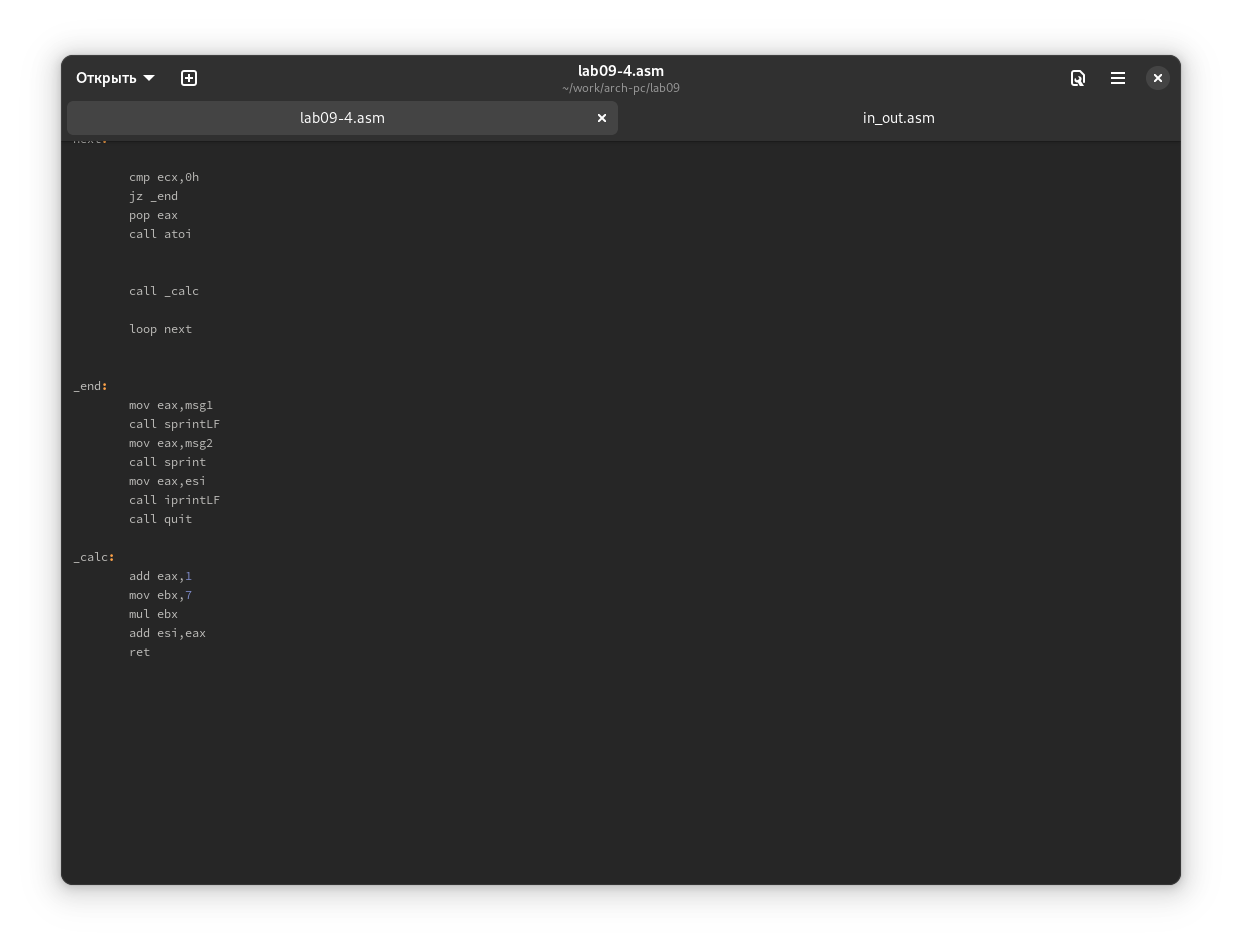
1. Смотрим позиции стека. Разница в 4 обусловлена размером каждой позиции в 4 байта.



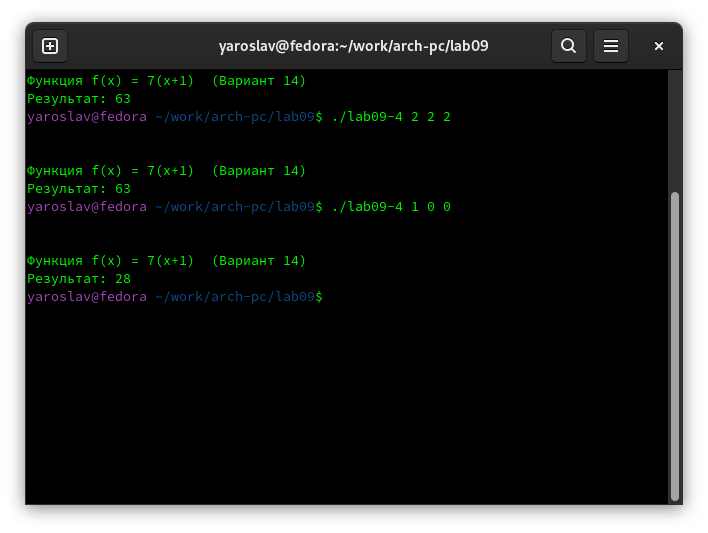
Просмотр элементов стека

## 2.3 Выполнение самостоятельной работы

1. Меняем текст предыдущего задания. Отправляем вычисление функции в отдельную подпрограмму \_calc.

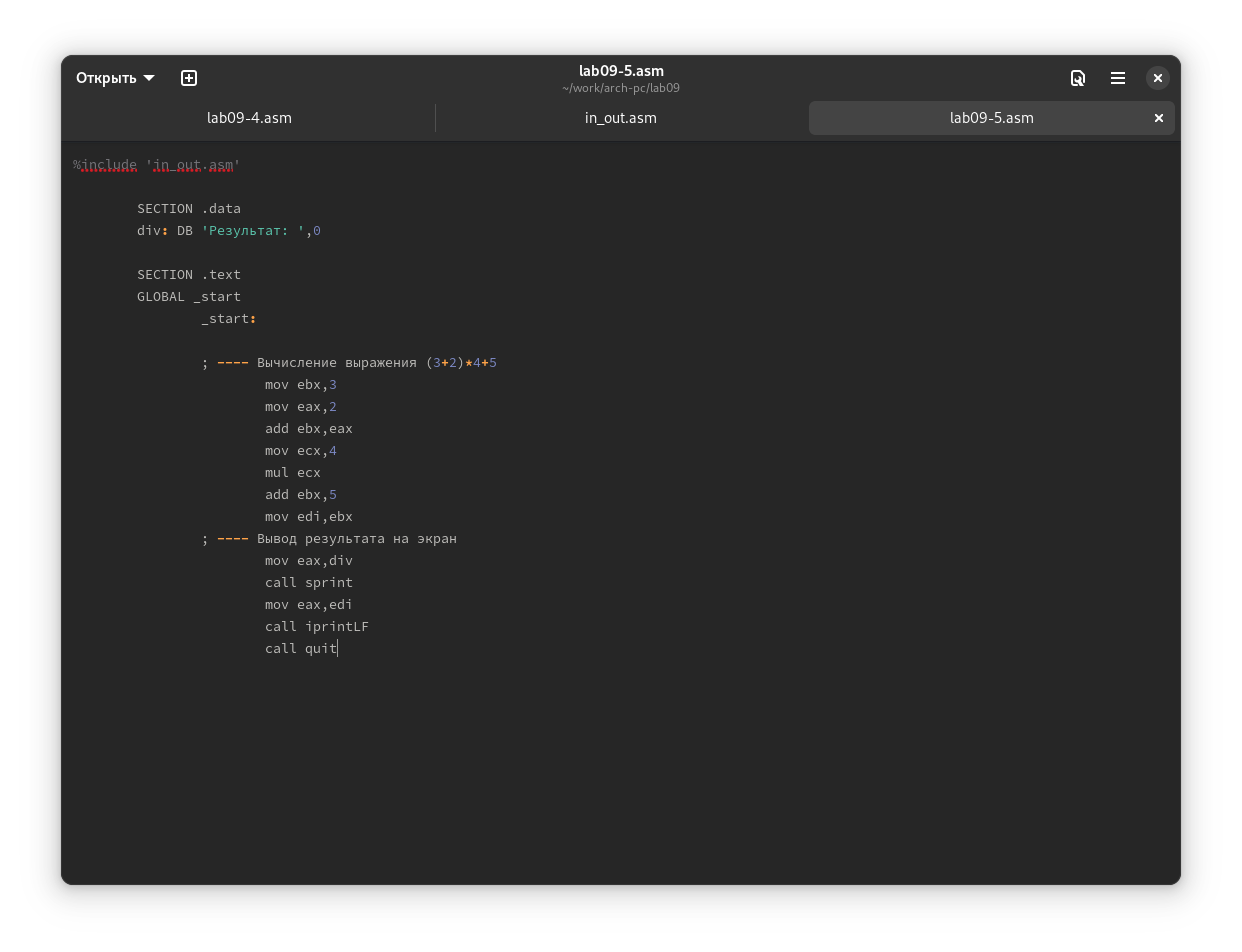


Измененная программа

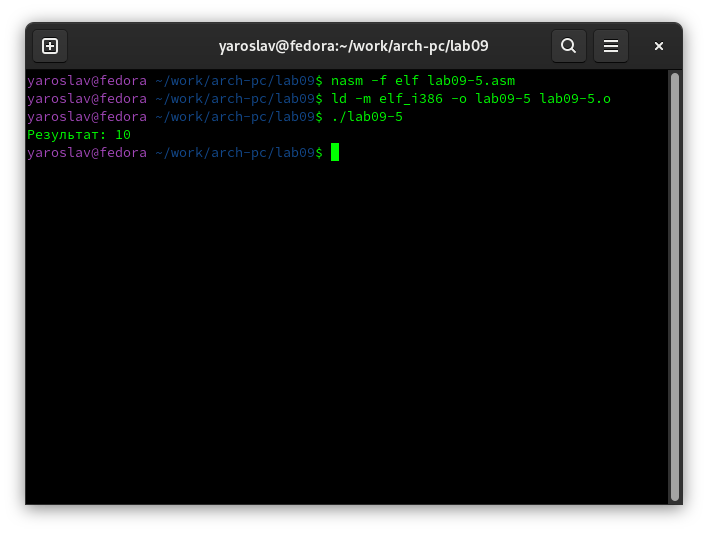


Работа программы

1. Пишем в файл заданный листинг.

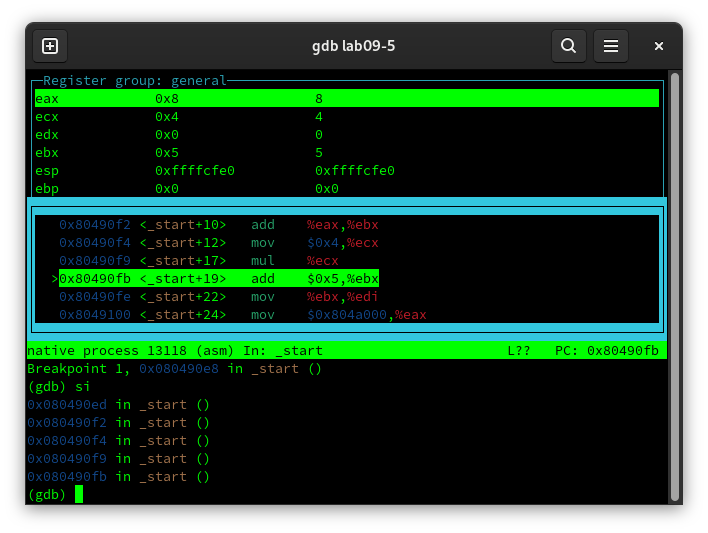


Заданный текст

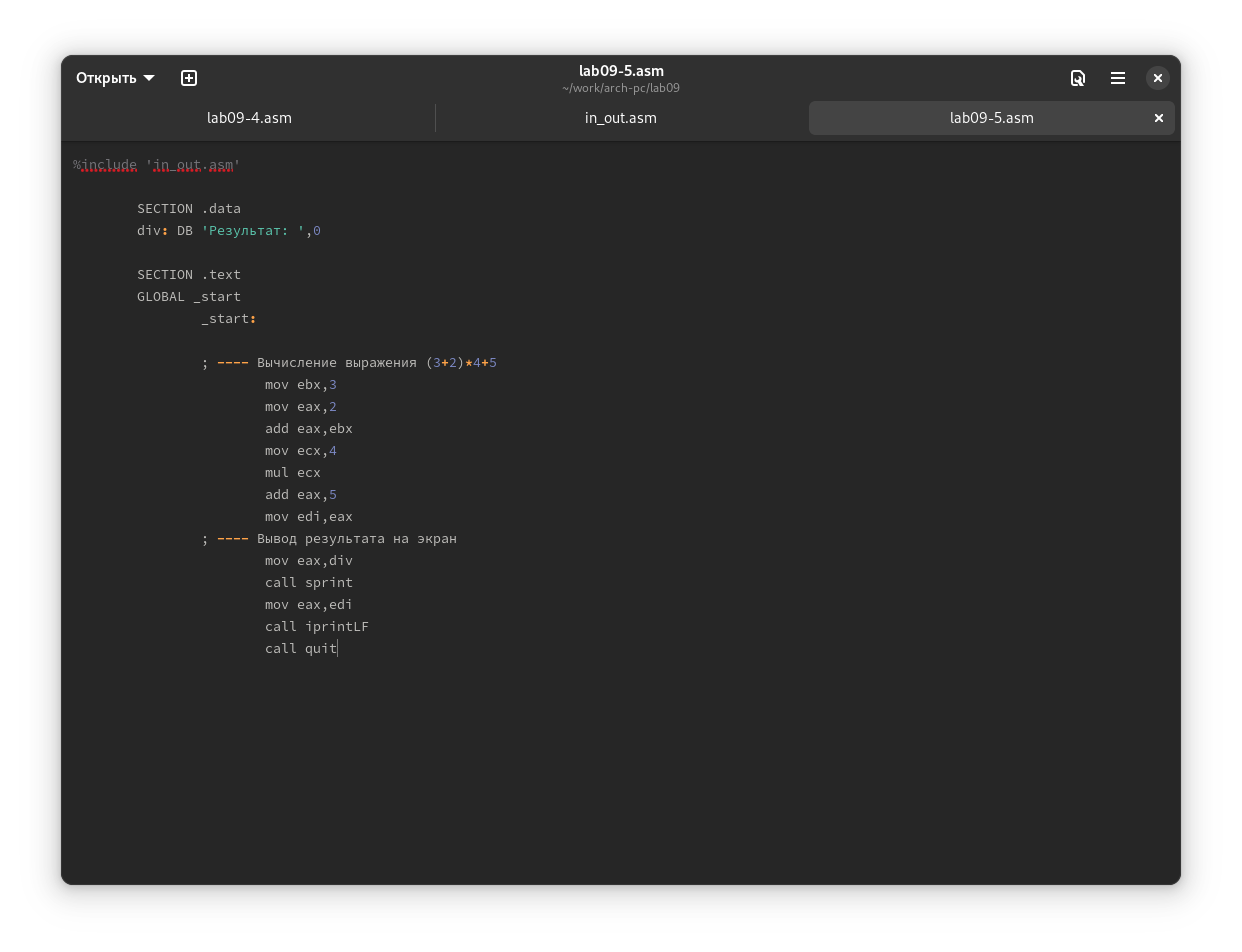


Неправильная работа программы

1. Запускаем отладчик и находим ошибку. Команда mul умножает на значение регистра и перезаписывает результат в eax (а не в ebx). Меняем ebx на eax.

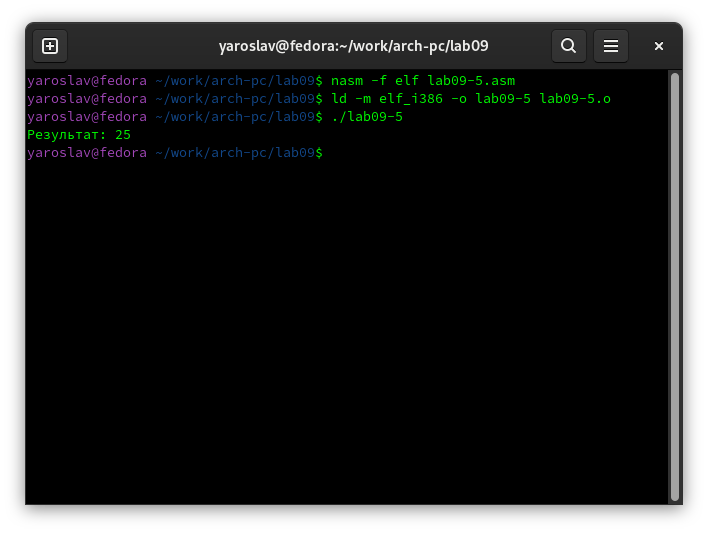


Проблема



Правильная программа

1. Проверяем работу.



Работа программы

# 3 Выводы

Были изучены подпрограммы и отладчик. Получены практические навыки их применений.