Отчёт по лабораторной работе №9

Дисциплина: архитектура компьютера.

Наговицын Арсений Владимирович

Содержание

# 1 Цель работы

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

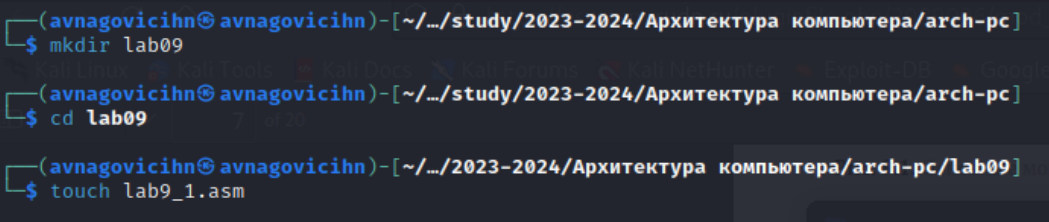
# 2 Задание

1. Реализация подпрограмм в NASM.
2. Отладка программам с помощью GDB.
3. Добавление точек останова.
4. Работа с данными программы в GDB.
5. Обработка аргументов командной строки в GDB.
6. Выполнение заданий для самостоятельной работы.

# 3 Выполнение лабораторной работы

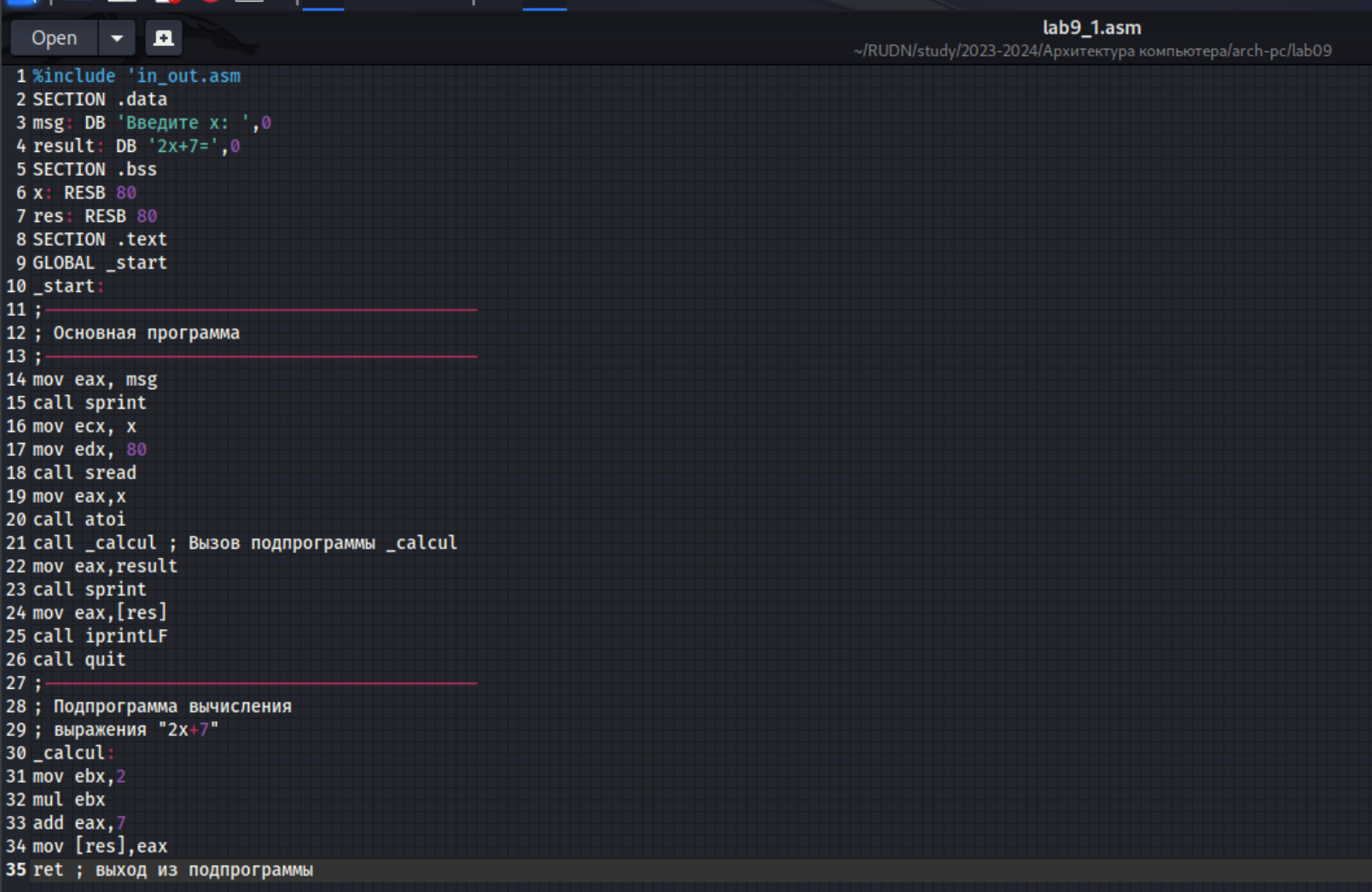
## 3.1 Реализация подпрограмм в NASM.

Создаю каталог для программ и перехожу в него. Создаю файл (рис. ??).



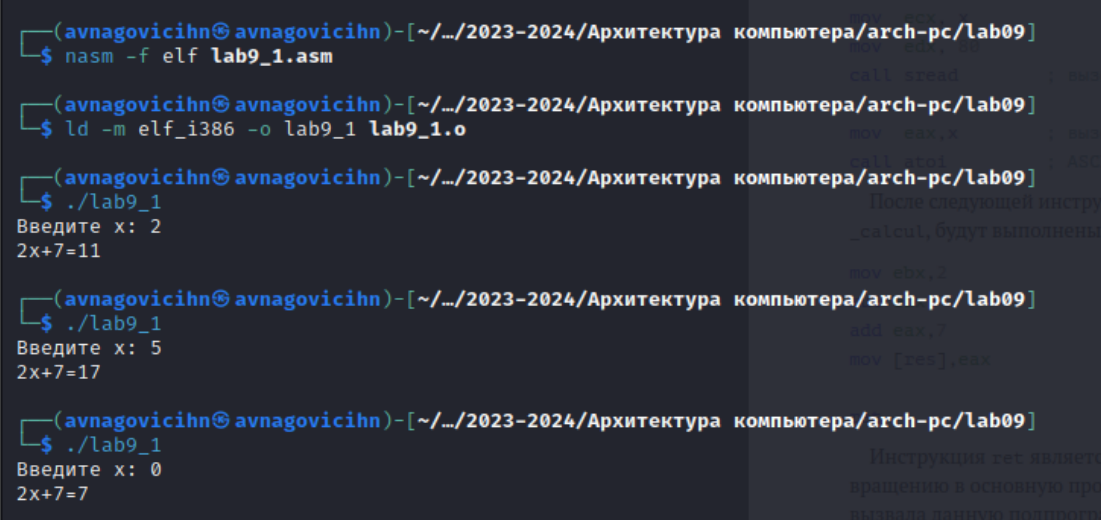
Создание каталога и файла

Ввожу в файл текст программы из листинга 9.1(рис. ??).



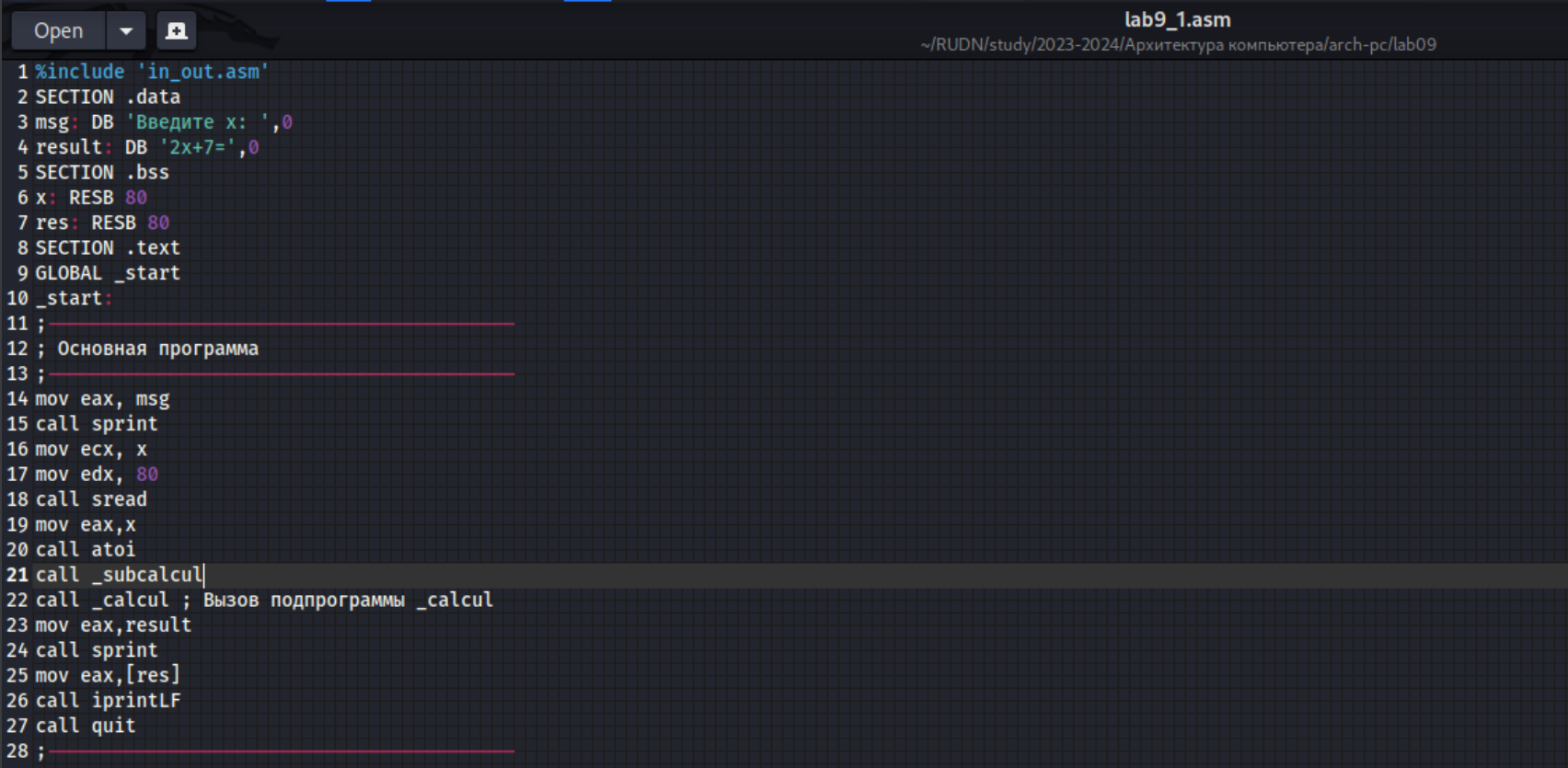
Редактирование файла

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



Компиляция и запуск файла

Изменяю текст программы, добавив подпрограмму \_subcalcul в подпрограмму \_calcul (рис. ??).

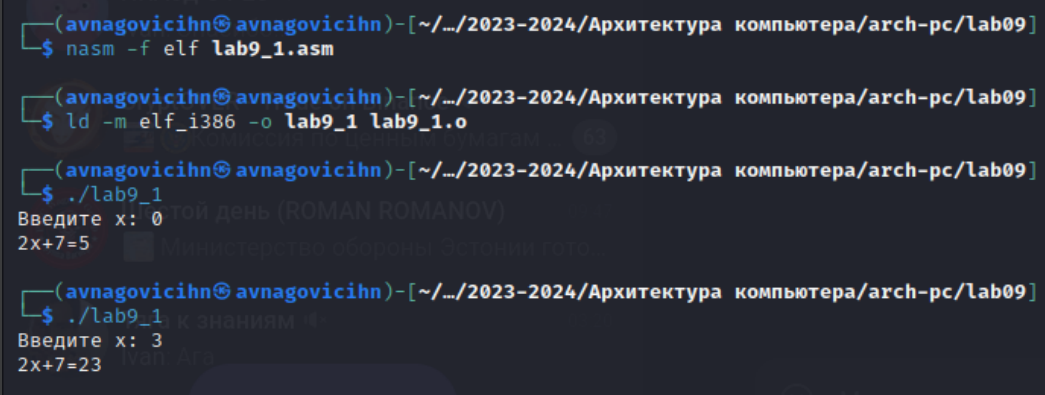


Редактирование файла

**Изменённая программа**

%include 'in\_out.asm'  
SECTION .data  
msg: DB 'Введите x: ',0  
result: DB '2x+7=',0  
SECTION .bss  
x: RESB 80  
res: RESB 80  
SECTION .text  
GLOBAL \_start  
\_start:  
;------------------------------------------  
; Основная программа  
;------------------------------------------  
mov eax, msg  
call sprint  
mov ecx, x  
mov edx, 80  
call sread  
mov eax,x  
call atoi  
call \_subcalcul  
call \_calcul ; Вызов подпрограммы \_calcul  
mov eax,result  
call sprint  
mov eax,[res]  
call iprintLF  
call quit  
;------------------------------------------  
; Подпрограмма вычисления  
; выражения "2x+7"  
\_subcalcul:  
mov ebx,3  
mul ebx  
add eax,-1  
ret  
\_calcul:  
mov ebx,2  
mul ebx  
add eax,7  
mov [res],eax  
ret ; выход из подпрограммы

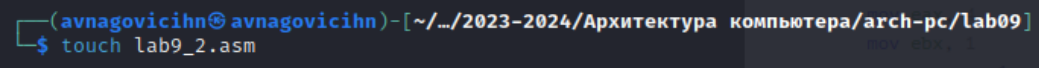
Создаю исполняемый файл и запускаю его (рис. ??).



Компиляция и запуск файла

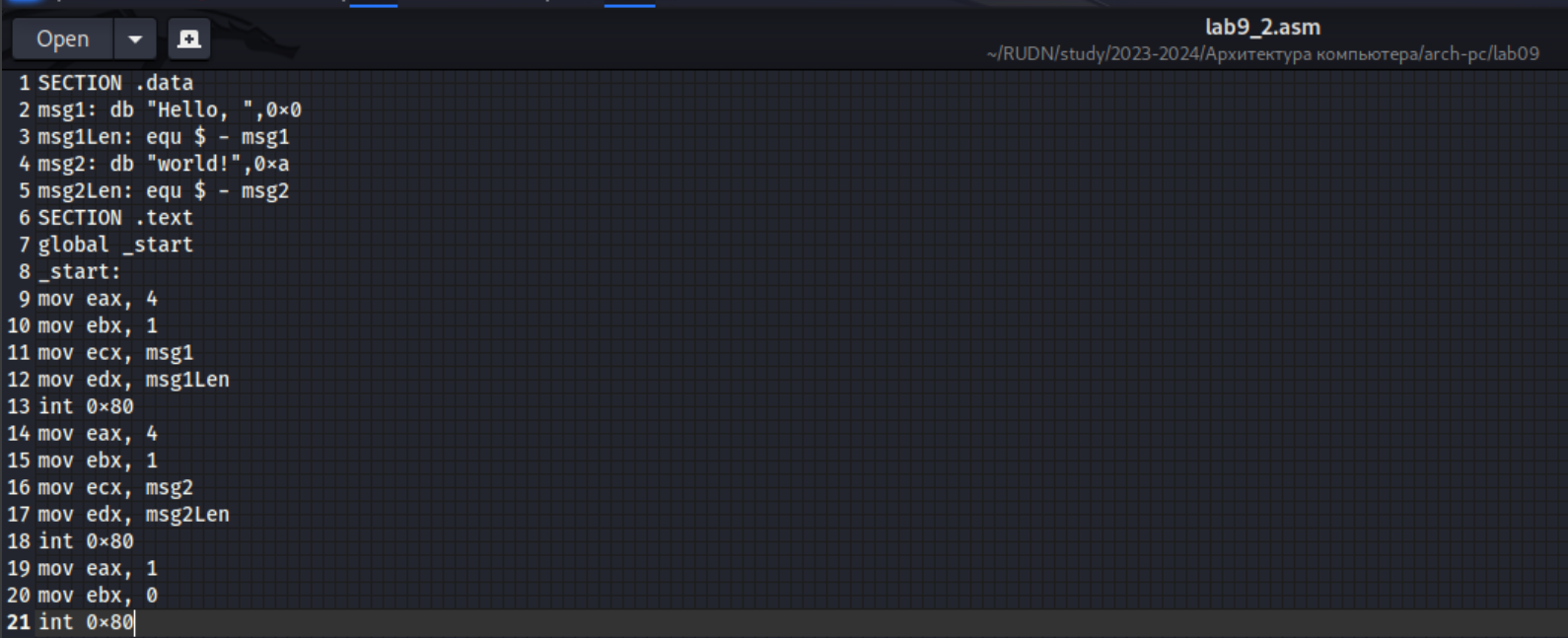
## 3.2 Отладка программам с помощью GDB.

Создаю файл lab09-2.asm (рис. ??).



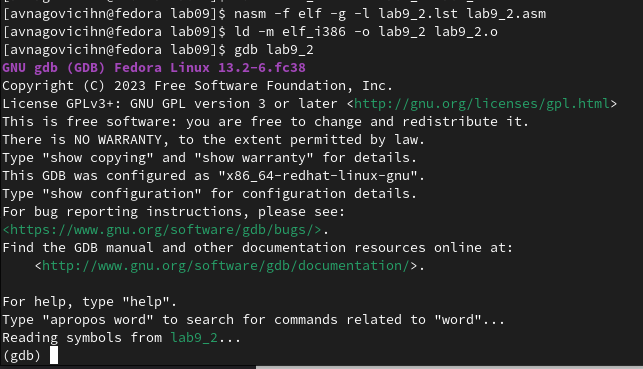
Создание файла

Ввожу в файл текст программы из листинга 9.1(рис. ??).



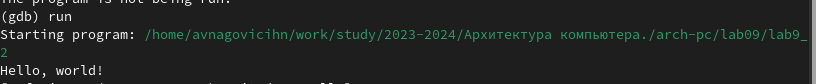
Редактирование файла

Создаю исполняемый файл и загружаю его в откладчик (рис. ??).



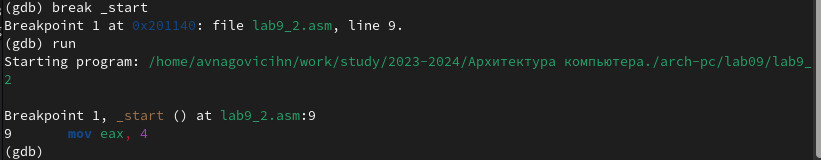
Компиляция и запуск файла

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



Запуск программы в отладчике

Ставлю точку останова на метку \_start и запустил её. (рис. ??).



Установка breakpoint

Просматриваю дисассимилированный код (рис. ??).



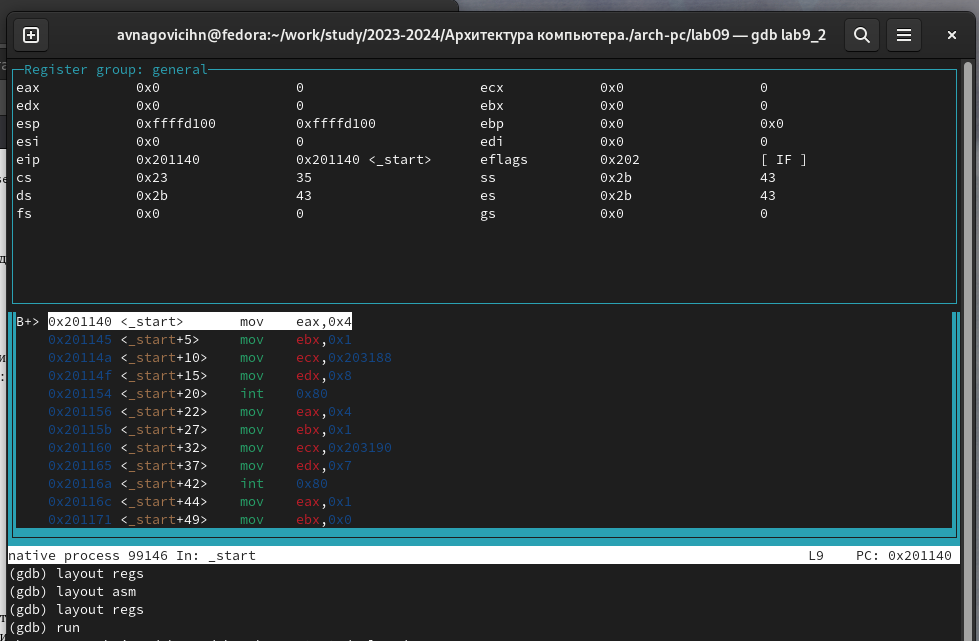
Дисассимилированный код

Просматриваю дисассимилированный код с Intel’овским синтаксисом (рис. ??).



Дисассимилированный код

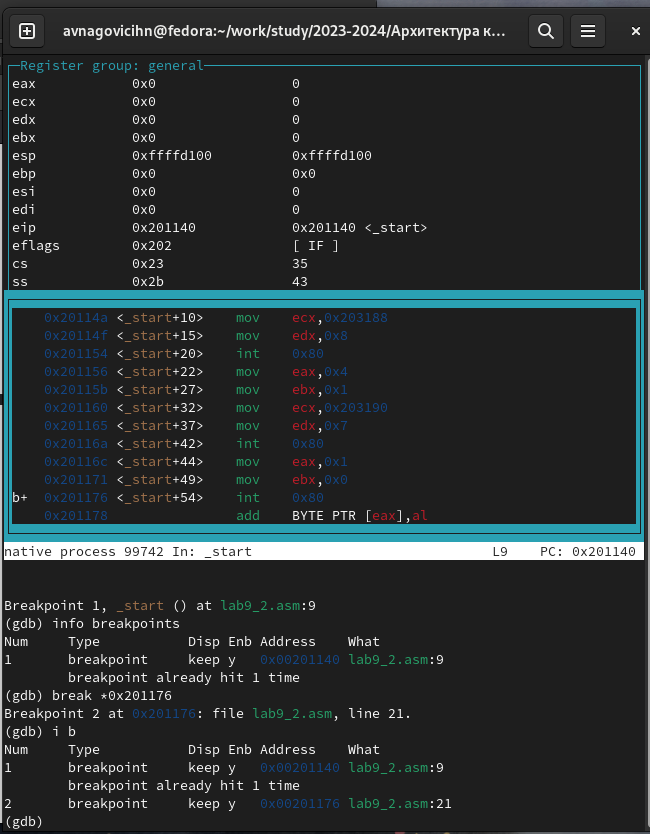
Включая режим графики (рис. ??).



Графика

## 3.3 Добавление точек останова

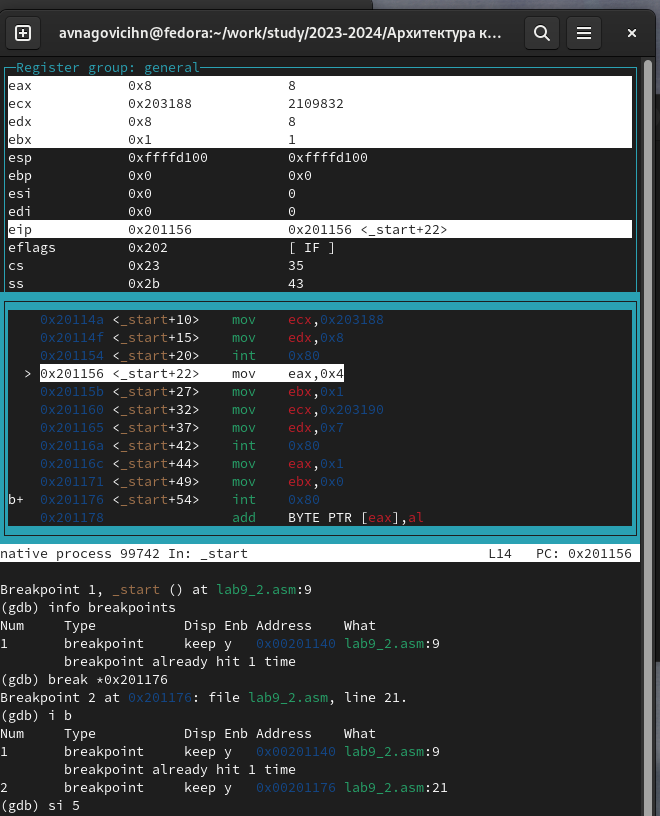
Проверяю установленные точки останова, добавляю еще одну точку останова((рис. ??).



Установленные точки останова

## 3.4 Работа с данными программы в GDB.

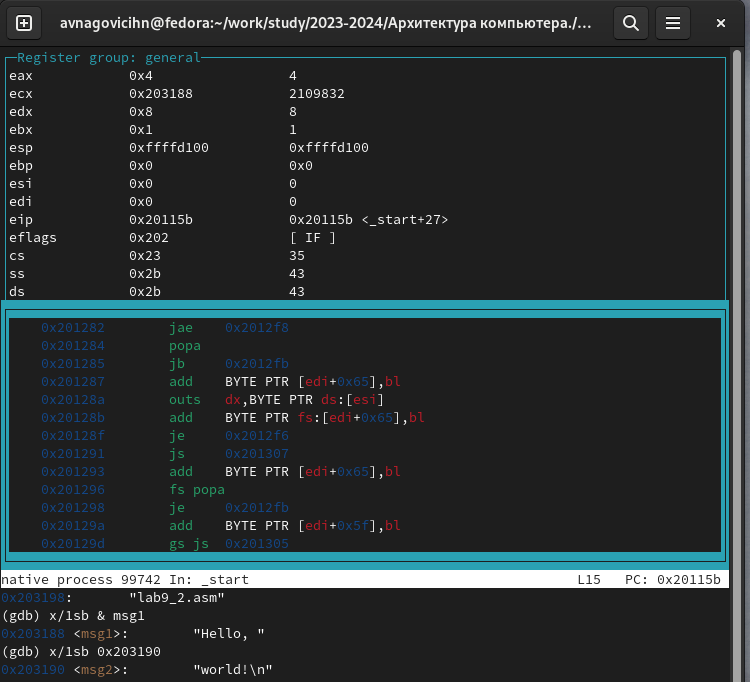
Выполнил 5 инструкций с помощью команды stepi (рис. ??).



Команды stepi

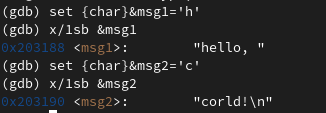
Изменились значения регистров eax, ecx, edx и ebx.

Посматриваю значение переменной msg1 по имени и msg2 по адресу (рис. ??).



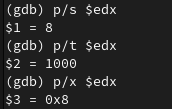
Значение переменной msg1

Изменяю первый символ переменных msg1 и msg2 (рис. ??).



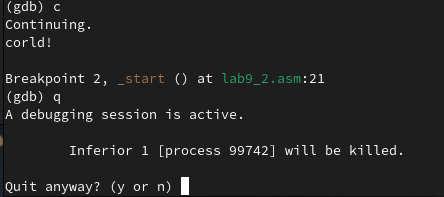
Команда set

Вывожу в различных форматах значение регистра edx (рис. ??).



Значение регистра edx

Завершаю выполнение программы с помощью команды continue и вquit (рис. ??).



Команда continue и quit

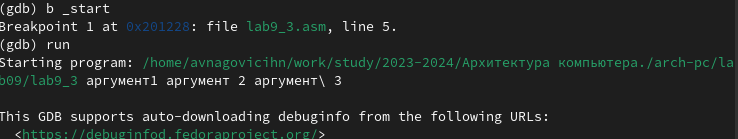
## 3.5 Обработка аргументов командной строки в GDB.

Скопировав файл и переименовав его. Создаю исполняемый файл (рис. ??).

Компиляция и запуск файла

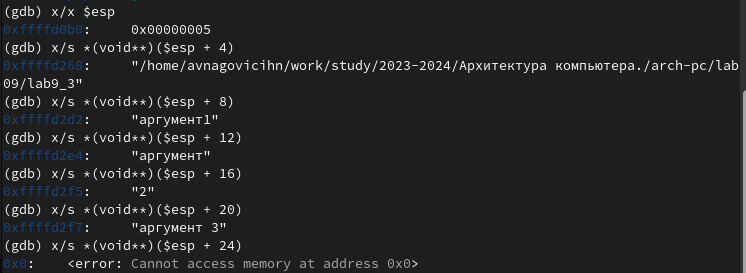
Компиляция и запуск файла

Устанавливаю точку останова и запускаю файл (рис. ??).



Установление тоски останова

Просматриваю вершину стека по их адресам (рис. ??).



Просмотр значений

Шаг изменения адреса равен 4, так как количество аргументов командной строки равно 4.

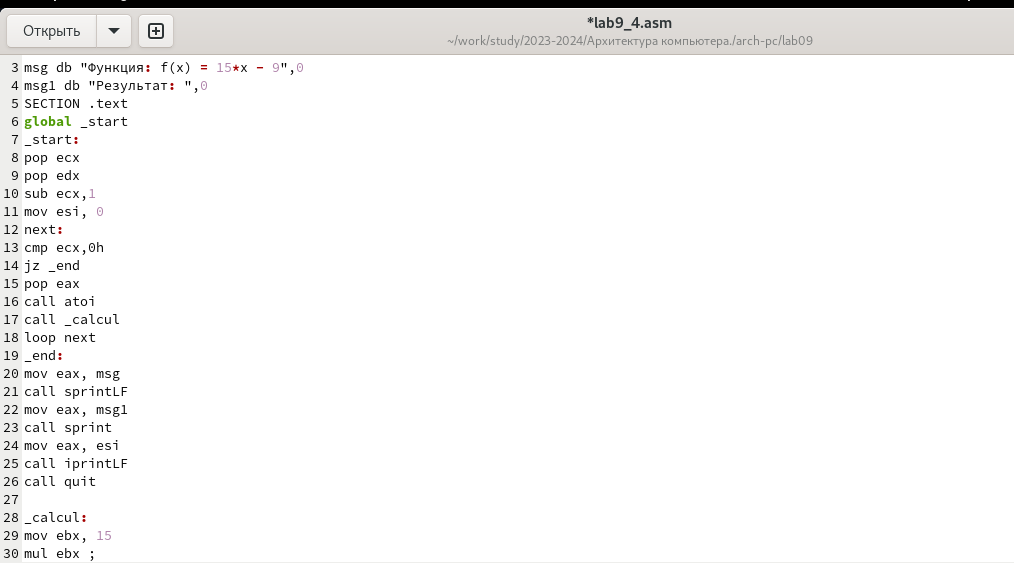
## 3.6 Выполнение заданий для самостоятельной работы.

1. Создаю файл (рис. ??).

Создание файла

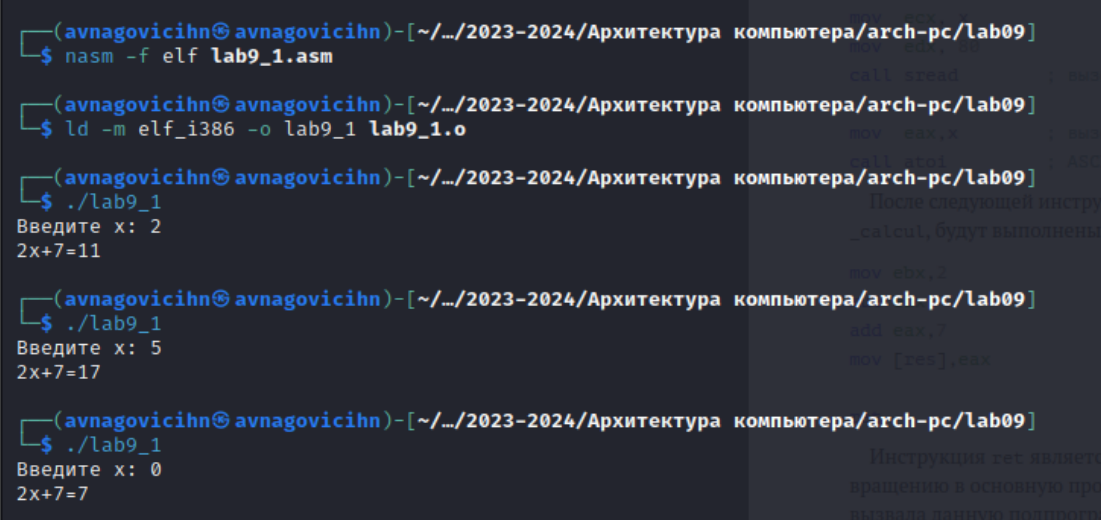
Создание файла

Ввожу в файл текст программы из листинга 9.4(рис. ??).

 **Листинг 9.4:**

%include 'in\_out.asm'   
SECTION .data   
msg db "Функция: f(x) = 15\*x - 9",0   
msg1 db "Результат: ",0   
SECTION .text   
global \_start   
\_start:   
pop ecx   
pop edx   
sub ecx,1   
mov esi, 0   
next:   
cmp ecx,0h   
jz \_end   
pop eax   
call atoi   
call \_calcul   
loop next   
\_end:   
mov eax, msg   
call sprintLF   
mov eax, msg1   
call sprint   
mov eax, esi   
call iprintLF   
call quit   
   
\_calcul:   
mov ebx, 15   
mul ebx ;   
add eax, -9 ;   
add esi, eax  
ret

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



Компиляция и запуск файла

1. Создаю файл lab9\_5.asm (рис. ??).

Создание файла

Создание файла

Ввожу в файл текст программы из листинга 9.3(рис. ??).



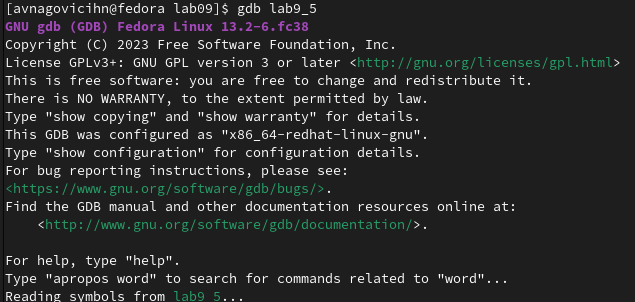
Редактирование файла

Создаю исполняемый файл (рис. ??).

Компиляция файла

Компиляция файла

Загружаю и запускаю файл программы в оболочке GDB (рис. ??).



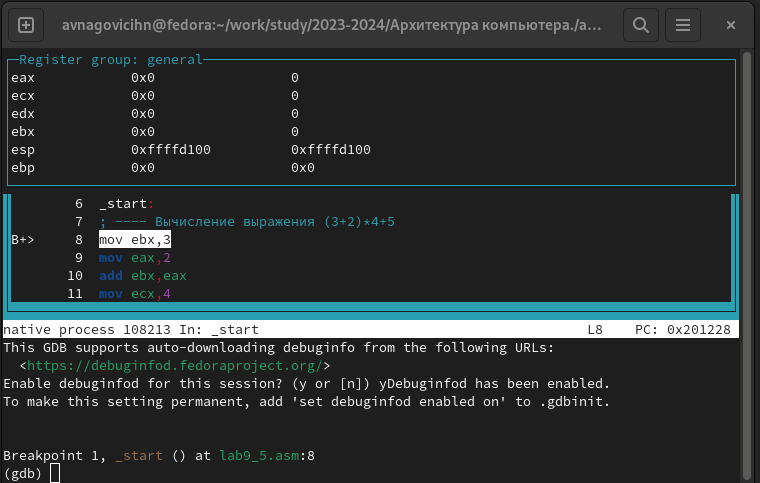
Программа в оболочке GDB

Ставлю точку Останова на метку \_start (рис. ??).

Программа в оболочке GDB

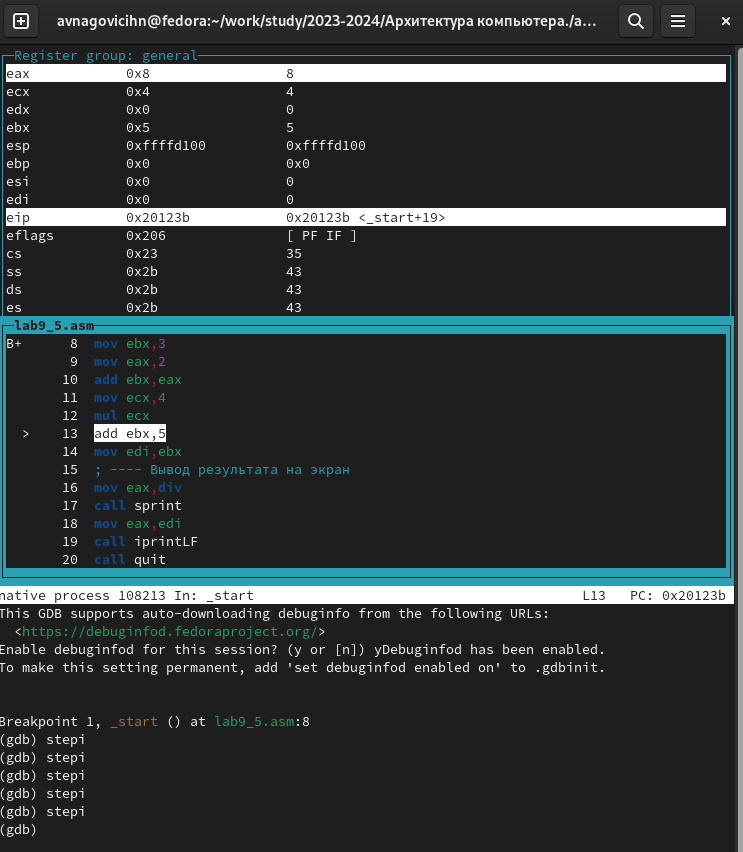
Программа в оболочке GDB

Включаю режим графики (рис. ??).



Команда layout

Ввожу команду stepi 5 раз, замечаю что умножается не тот регистр (рис. ??).



Команды stepi

Изменяю текст программы. (рис. ??).

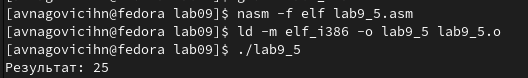


Редактирование файла

**Листинг 9.5:**

%include 'in\_out.asm'  
SECTION .data  
div: DB 'Результат: ',0  
SECTION .text  
GLOBAL \_start  
\_start:  
; ---- Вычисление выражения (3+2)\*4+5  
mov ebx,3  
mov eax,2  
add eax,ebx  
mov ecx,4  
mul ecx  
add eax,5  
mov edi,eax  
; ---- Вывод результата на экран  
mov eax,div  
call sprint  
mov eax,edi  
call iprintLF  
call quit

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

 # Выводы

В этой лабораторной работе были приобретены навыки написания программ с использованием подпрограмм, а также были рассмотрены методы отладки при помощи GDB и его основные возможности.