

Лабораторная работа №9. Понятие подпрограммы. Отладчик GDB.

НПМбВ-01-21

Ермаков Алексей

Содержание

Цель работы	5
Задание	6
Выполнение лабораторной работы	7
Выводы	14

Список таблиц

Список иллюстраций

1	Создание и проверка работы файла вычисления арифметического выражения	7
2	Пример работы файла с двумя подпрограммами вычисления арифметического выражения	7
3	Создание и проверка работы файла печати сообщения и запуск отладки	8
4	Создание и проверка работы файла вычисления арифметического выражения	8
5	Установил точка останова по адресу инструкции в режиме псевдографики	9
6	Посмотр значение переменной по адресу используя отображения содержимого памяти	10
7	Посмотр позиции стека и определение размера шага изменения адреса.	11
8	Преобразовали программу из лабораторной работы №8, реализовав вычисление значения функции как подпрограмму	11
9	Создали файл вычисления арифметического выражения, проверили его работу, обратили внимание на ошибку и запустили отладку. . . .	12
10	Определение ошибки с помощью отладчика GDB	12
11	Создание и проверка работы исправленного файла вычисления арифметического выражения	13

Цель работы

Получить навыки написания программ с использованием подпрограмм. Познакомиться с методами отладки при помощи GDB и его основными возможностями.

Задание

1. Преобразуйте программу из лабораторной работы №8 (Задание №1 для самостоятельной работы), реализовав вычисление значения функции как подпрограмму.
2. В листинге 9.3 приведена программа вычисления выражения. При запуске данная программа дает неверный результат. Проверьте это. С помощью отладчика GDB, анализируя изменения значений регистров, определите ошибку и исправьте ее

Выполнение лабораторной работы

Создал каталог для программ лабораторной работы, написал в файл lab09-1.asm текст программы из листинга 9.1. Создал исполняемый файл и проверил его работу. (рис. @fig:001).

```
avermakov@avermakov-VirtualBox:~/work/lab09$ nasm -f elf lab09-1.asm
avermakov@avermakov-VirtualBox:~/work/lab09$ ld -m elf_i386 lab09-1.o -o lab09-1
avermakov@avermakov-VirtualBox:~/work/lab09$ ./lab09-1
Введите x: 3
2x+7=13
avermakov@avermakov-VirtualBox:~/work/lab09$
```

Рис. 1: Создание и проверка работы файла вычисления арифметического выражения

Изменил текст программы добавив две подпрограммы вычисления арифметического выражения. Создал исполняемый файл и проверил его работу(рис. @fig:002).

```
avermakov@avermakov-VirtualBox:~/work/lab09$ ./lab09-1
Введите x: 3
2x+7=13
avermakov@avermakov-VirtualBox:~/work/lab09$ nasm -f elf lab09-1.asm
avermakov@avermakov-VirtualBox:~/work/lab09$ ld -m elf_i386 lab09-1.o -o lab09-1
avermakov@avermakov-VirtualBox:~/work/lab09$ ./lab09-1
f(x) = 2x+7
g(x)= 3x-1
Введите x: 3
f(g(x))=23
avermakov@avermakov-VirtualBox:~/work/lab09$
```

Рис. 2: Пример работы файла с двумя подпрограммами вычисления арифметического выражения

Создал файл lab09-2.asm с текстом программы из Листинга 9.2. (Программа печати сообщения Hello world!) и проверил его работу. Начал отладку(рис. @fig:003).

```

avermakov@avermakov-VirtualBox:~/work/lab09$ touch lab09-2.asm
avermakov@avermakov-VirtualBox:~/work/lab09$ nasm -f elf -g -l lab09-2.lst lab09-2.asm
avermakov@avermakov-VirtualBox:~/work/lab09$ ld -m elf_i386 -o lab09-2 lab09-2.o
avermakov@avermakov-VirtualBox:~/work/lab09$ gdb lab09-2
GNU gdb (Ubuntu 12.1-0ubuntu1~22.04) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from lab09-2...
(gdb)

```

Рис. 3: Создание и проверка работы файла печати сообщения и запуск отладки

Проверил работу программы, запустив ее в оболочке GDB с помощью команды `run`. (рис. @fig:004).

```

Reading symbols from lab09-2...
(gdb) run
Starting program: /home/avermakov/work/lab09/lab09-2
Hello, world!
[Inferior 1 (process 19519) exited normally]
(gdb)

```

Рис. 4: Создание и проверка работы файла вычисления арифметического выражения

В режиме псевдографики `gdb` была установлена точка останова по адресу инструкции. (рис. @fig:005).


```
avermakov@avermakov-VirtualBox: ~/work/lab09
Файл  Правка  Вид  Поиск  Терминал  Справка

[ Register Values Unavailable ]

b+ 0x8049000 <_start>    mov    eax,0x4
0x8049005 <_start+5>    mov    ebx,0x1
0x804900a <_start+10>   mov    ecx,0x804a000
0x804900f <_start+15>   mov    edx,0x8
0x8049014 <_start+20>   int     0x80
0x8049016 <_start+22>   mov    eax,0x4
0x804901b <_start+27>   mov    ebx,0x1

exec No process In:
(gdb) i b
Num   Type           Disp Enb Address      What
1     breakpoint      keep y   0x08049000  lab09-2.asm:9
2     breakpoint      keep y   <PENDING>  0x8049000
(gdb) i r
The program has no registers now.
(gdb) 
```

Рис. 5: Установил точка останова по адресу инструкции в режиме псевдографики

Посмотрели значение переменной по адресу используя отображения содержимого памяти. Посмотрели инструкцию `mov ecx,msg2` которая записывает в регистр `ecx` адрес переменной `msg2` (рис. @fig:006).

```
avermakov@avermakov-VirtualBox: ~/work/lab09
Файл  Правка  Вид  Поиск  Терминал  Справка

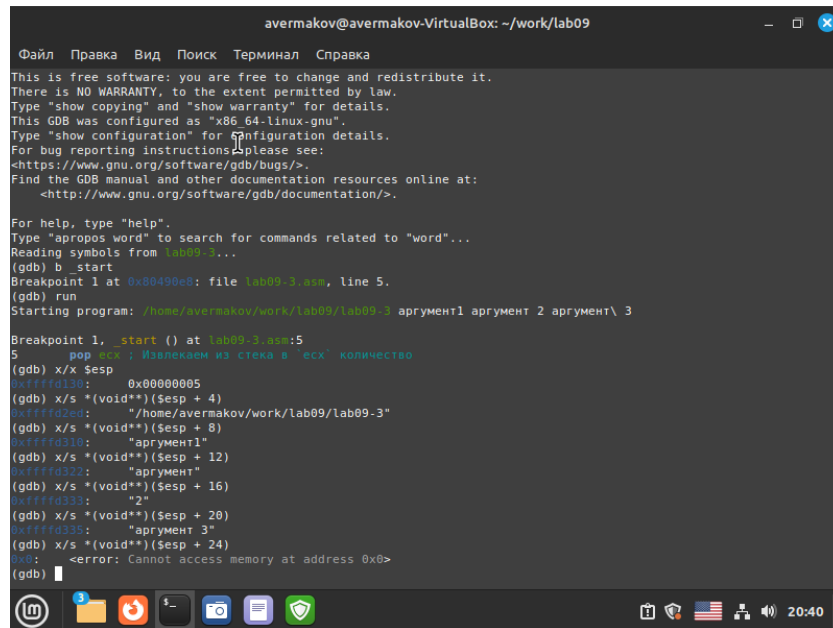
[ Register Values Unavailable ]

b+  0x8049000 <_start>    mov    eax,0x4
    0x8049005 <_start+5>   mov    ebx,0x1
    0x804900a <_start+10>  mov    ecx,0x804a000
    0x804900f <_start+15>  mov    edx,0x8
    0x8049014 <_start+20>  int     0x80
    0x8049016 <_start+22>  mov    eax,0x4
    0x804901b <_start+27>  mov    ebx,0x1

exec No process In:
0x804a000 <msg1>:      "Hello, "
(gdb) set {char}msg1='h'
'msg1' has unknown type; cast it to its declared type
(gdb) x/1sb &msg1
0x804a000 <msg1>:      "Hello, "
(gdb) set {char}&msg1='h'
Cannot access memory at address 0x804a000
(gdb) x/1sb &msg1
0x804a000 <msg1>:      "Hello, "
(gdb) 
```

Рис. 6: Посмотр значение переменной по адресу используя отображения содержимого памяти

Посмотр позиции стека. Размер переменной - четыре байта и шаг изменения адреса равен размеру переменной (рис. @fig:007).



```

avermakov@avermakov-VirtualBox: ~/work/lab09
Файл  Правка  Вид  Поиск  Терминал  Справка
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

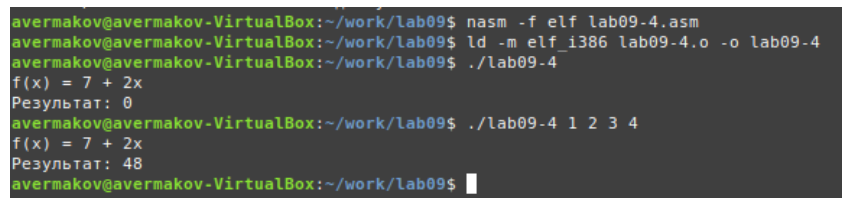
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from lab09-3...
(gdb) b _start
Breakpoint 1 at 0x80490e8: file lab09-3.asm, line 5.
(gdb) run
Starting program: /home/avermakov/work/lab09/lab09-3 аргумент1 аргумент2 аргумент\ 3

Breakpoint 1, _start () at lab09-3.asm:5
5      pop еск ; Извлекаем из стека в 'еск' количество
(gdb) x/x $esp
0xffffd130:  0x00000005
(gdb) x/s *(void**)(esp + 4)
0xffffd0d0:  "/home/avermakov/work/lab09/lab09-3"
(gdb) x/s *(void**)(esp + 8)
0xffffd010:  "аргумент1"
(gdb) x/s *(void**)(esp + 12)
0xffffd322:  "аргумент"
(gdb) x/s *(void**)(esp + 16)
0xffffd333:  "2"
(gdb) x/s *(void**)(esp + 20)
0xffffd335:  "аргумент 3"
(gdb) x/s *(void**)(esp + 24)
gdb:  <error: Cannot access memory at address 0x0>
(gdb)

```

Рис. 7: Посмотр позиции стека и определение размера шага изменения адреса.

Преобразовали программу из лабораторной работы №8 (Задание №1 для самостоятельной работы), реализовав вычисление значения функции как подпрограмму. (рис. @fig:008).



```

avermakov@avermakov-VirtualBox:~/work/lab09$ nasm -f elf lab09-4.asm
avermakov@avermakov-VirtualBox:~/work/lab09$ ld -m elf_i386 lab09-4.o -o lab09-4
avermakov@avermakov-VirtualBox:~/work/lab09$ ./lab09-4
f(x) = 7 + 2x
Результат: 0
avermakov@avermakov-VirtualBox:~/work/lab09$ ./lab09-4 1 2 3 4
f(x) = 7 + 2x
Результат: 48
avermakov@avermakov-VirtualBox:~/work/lab09$

```

Рис. 8: Преобразовали программу из лабораторной работы №8, реализовав вычисление значения функции как подпрограмму

Создали файл вычисления арифметического выражения, проверили его работу, обратили внимание на ошибку и запустили отладку. (рис. @fig:009).

```

Файл  Правка  Вид  Поиск  Терминал  Справка
f(x) = 7 + 2x
Результат: 0
avermakov@avermakov-VirtualBox: ~/work/lab09$ ./lab09-4 1 2 3 4
f(x) = 7 + 2x
Результат: 48
avermakov@avermakov-VirtualBox: ~/work/lab09$ touch lab09-5.asm
avermakov@avermakov-VirtualBox: ~/work/lab09$ nasm -f elf lab09-5.asm
avermakov@avermakov-VirtualBox: ~/work/lab09$ ld -m elf_i386 lab09-5.o -o lab09-5
avermakov@avermakov-VirtualBox: ~/work/lab09$ ./lab09-5
Результат: 10
avermakov@avermakov-VirtualBox: ~/work/lab09$ gdb lab09-5
GNU gdb (Ubuntu 12.1-0ubuntu1-22.04) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from lab09-5...
(No debugging symbols found in lab09-5)
(gdb) b _start
Breakpoint 1 at 0x80490e8
(gdb) run
Starting program: /home/avermakov/work/lab09/lab09-5

Breakpoint 1, 0x80490e8 in _start ()
(gdb)

```

Рис. 9: Создали файл вычисления арифметического выражения, проверили его работу, обратили внимание на ошибку и запустили отладку.

С помощью отладчика GDB, анализируя изменения значений регистров, определил ошибку и исправил(рис. @fig:0010).

```

Файл  Правка  Вид  Поиск  Терминал  Справка
--Register group: general--
eax      0x0      8
ecx      0x4      4
edx      0x0      0
ebx      0xa      10
esp      0xffffd180 0xffffd180
ebp      0x0      0x0
esi      0x0      0
edi      0x0      0
eip      0x80490fe 0x80490fe < start+22>

0+ 0x80490e8 < start> mov     ebx,0x3
0x80490ed < start+5> mov     eax,0x2
0x80490f2 < start+10> add     ebx,eax
0x80490f4 < start+12> mov     ecx,0x4
0x80490f9 < start+17> mul     ecx
0x80490fb < start+19> add     ebx,0x5
> 0x80490fe < start+22> mov     edi,ebx
0x8049100 < start+24> mov     eax,0x804a000
0x8049105 < start+29> call    0x804900f <sprint>
0x804910a < start+34> mov     eax,edi

native process 21632 In: start
L77 PC: 0x80490fe
(gdb) si
0x80490e8 in _start ()
(gdb) si
0x80490f2 in _start ()
(gdb) si
0x80490f4 in _start ()
(gdb) si
0x80490f9 in _start ()
(gdb) si
0x80490fb in _start ()
(gdb) si
0x80490fe in _start ()
(gdb)

```

Рис. 10: Определение ошибки с помощью отладчика GDB

Создал исполняемый исправленный файл, проверил его работу и убедился, что работает скрипт корректно. (рис. @fig:0011).

```
END of assembler dump.  
(gdb) layout asm  
avermakov@avermakov-VirtualBox:~/work/lab09$ nasm -f elf lab09-5.asm  
avermakov@avermakov-VirtualBox:~/work/lab09$ ld -m elf_i386 lab09-5.o -o lab09-5  
avermakov@avermakov-VirtualBox:~/work/lab09$ ./lab09-5  
Результат: 25  
avermakov@avermakov-VirtualBox:~/work/lab09$
```

Рис. 11: Создание и проверка работы исправленного файла вычисления арифметического выражения

Выводы

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