Отчет по лабораторной работа №9

Дисциплина: архитектура компьютера

Ширинкин Т. Б.

Содержание

Цель работы

Приобретение навыков программирования с использованием подпрограмм, знакомство с отладкой при помощи gdb.

Выполнение лабораторной работы

Создал каталог и первый файл (Рис. [-@fig:000])

```
tbshirinkin@Shiza:~$ mkdir ~/work/arch-pc/lab09
tbshirinkin@Shiza:~$ cd ~/work/arch-pc/lab09
tbshirinkin@Shiza:~/work/arch-pc/lab09$ touch lab09-1.asm
```

Рис. 0 Создал каталог и первый файл

Ввёл первый листинг, изменив его с нужнми исправлениями (Рис. [-@fig:001])

```
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 ; Вызов подпрограммы _subcalcul
mov eax,result
call _calcul ; Вызов подпрограммы _calcul
mov eax,result
call sprint
mov eax,[res]
call iprintLF
call quit
;---------------------------------------------
; Подпрограмма вычисления
; выражения "2x+7"
_calcul:
mov ebx,2
mul ebx
add eax,7
mov [res],eax
ret ; выход из подпрограммы
;---------------------------------------------
; Подпрограмма вычисления
; выражения "3x-1"
_subcalcul:
mov ebx,3
mul ebx
sub eax,1
mov [res],eax
ret ; выход из подпрограммы
                                              46,1          Bot
```

Рис. 1 Ввёл первый листинг, изменив его с нужнми исправлениями

Создал исполняемый файл и запустил его : работает (Рис. [-@fig:002])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf lab09-1.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-1 lab09-1.o
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ./lab09-1
Введите x: 7
2(3x-1)+7=47
```

Рис. 2 Создал исполняемый файл и запустил его

Копируем новый листинг и компилируем , запускаем gdb, вводим все команды (Рис. [-@fig:003])(Рис. [-@fig:004]) (Рис. [-@fig:005]) (Рис. [-@fig:006]) (Рис. [-@fig:007])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf -g -l lab09-2.lst lab09-2.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-2 lab09-2.o
tbshirinkin@Shiza:~/work/arch-pc/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/>.
```

Рис. 3 Копируем новый листинг и компилируем

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from lab09-2...
(gdb) run
Starting program: /home/tbshirinkin/work/arch-pc/lab09/lab09-2
Hello, world!
[Inferior 1 (process 981) exited normally]
(gdb)  break _start
Breakpoint 1 at 0x8049000: file lab09-2.asm, line 9.
(gdb) run
Starting program: /home/tbshirinkin/work/arch-pc/lab09/lab09-2

Breakpoint 1, _start () at lab09-2.asm:9
9       mov eax, 4
(gdb) disassemble _start
Dump of assembler code for function _start:
=> 0x08049000 <+0>:     mov    $0x4,%eax
   0x08049005 <+5>:     mov    $0x1,%ebx
   0x0804900a <+10>:    mov    $0x804a000,%ecx
   0x0804900f <+15>:    mov    $0x8,%edx
   0x08049014 <+20>:    int    $0x80
   0x08049016 <+22>:    mov    $0x4,%eax
   0x0804901b <+27>:    mov    $0x1,%ebx
   0x08049020 <+32>:    mov    $0x804a008,%ecx
   0x08049025 <+37>:    mov    $0x7,%edx
   0x0804902a <+42>:    int    $0x80
   0x0804902c <+44>:    mov    $0x1,%eax
   0x08049031 <+49>:    mov    $0x0,%ebx
   0x08049036 <+54>:    int    $0x80
End of assembler dump.
(gdb) set disassembly-flavor intel
(gdb) disassemble _start
Dump of assembler code for function _start:
=> 0x08049000 <+0>:     mov    eax,0x4
   0x08049005 <+5>:     mov    ebx,0x1
   0x0804900a <+10>:    mov    ecx,0x804a000
   0x0804900f <+15>:    mov    edx,0x8
   0x08049014 <+20>:    int    0x80
   0x08049016 <+22>:    mov    eax,0x4
   0x0804901b <+27>:    mov    ebx,0x1
   0x08049020 <+32>:    mov    ecx,0x804a008
   0x08049025 <+37>:    mov    edx,0x7
   0x0804902a <+42>:    int    0x80
   0x0804902c <+44>:    mov    eax,0x1
   0x08049031 <+49>:    mov    ebx,0x0
   0x08049036 <+54>:    int    0x80
End of assembler dump.
(gdb) layout asm
```

Рис. 4 Копируем новый листинг и компилируем

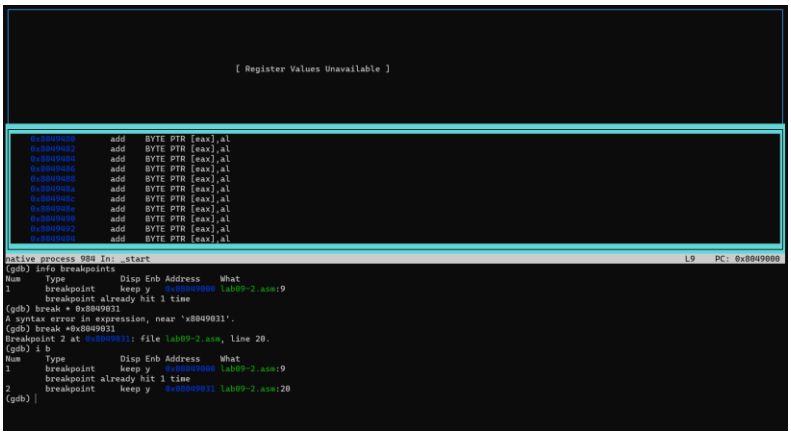

Рис. 5 Копируем новый листинг и компилируем

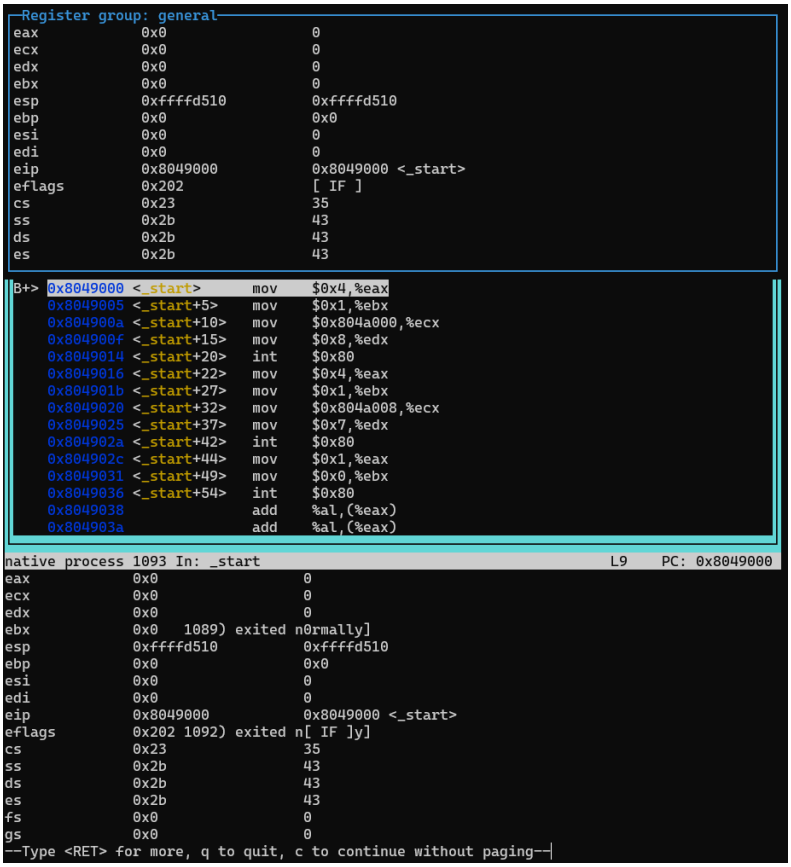

Рис. 6 Копируем новый листинг и компилируем

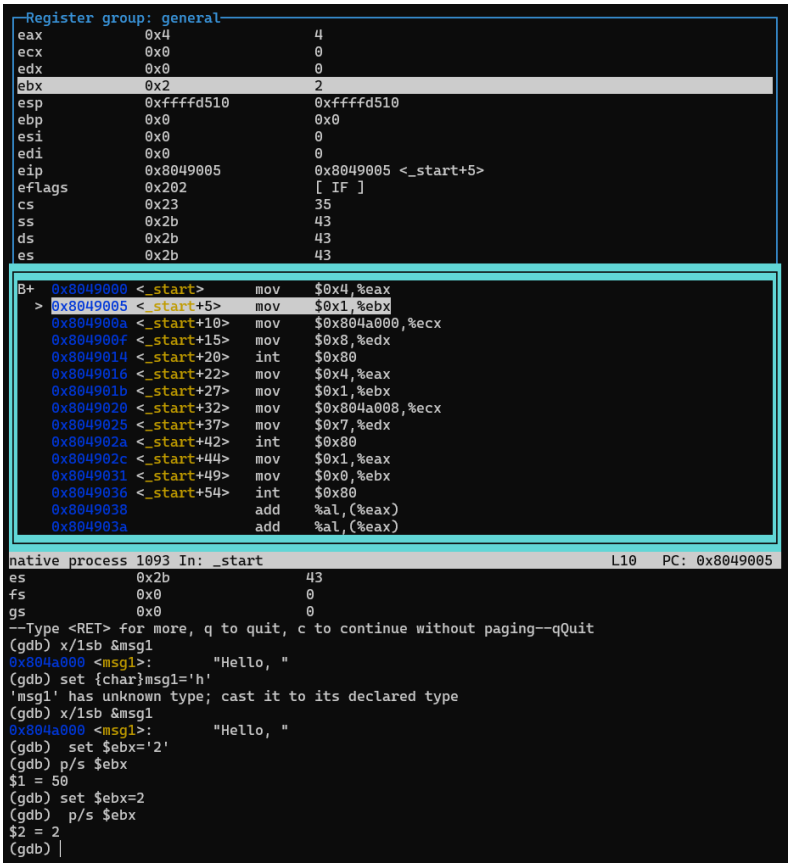

Рис. 7 Копируем новый листинг и компилируем

Копируем файл и отлаживаем : в стеке всё как в прошлой лабораторной говорилось (Рис. [-@fig:008])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ cp ~/work/arch-pc/lab08/lab8-2.asm ~/work/arch-pc/l
ab09/lab09-3.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf -g -l lab09-3.lst lab09-3.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-3 lab09-3.o
tbshirinkin@Shiza:~/work/arch-pc/lab09$ gdb --args lab09-3 аргумент1 аргумент 2 'аргумент 3
'
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 lab09-3...
(gdb) b _start
Breakpoint 1 at 0x80490e8: file lab09-3.asm, line 5.
(gdb) run
Starting program: /home/tbshirinkin/work/arch-pc/lab09/lab09-3 аргумент1 аргумент 2 аргумен
т\ 3

Breakpoint 1, _start () at lab09-3.asm:5
5       pop ecx ; Извлекаем из стека в `ecx` количество
(gdb) x/x $esp
0xffffd4d0:     0x00000005
(gdb) x/s *(void**)($esp + 4)
0xffffd644:     "/home/tbshirinkin/work/arch-pc/lab09/lab09-3"
(gdb) x/s *(void**)($esp + 8)
0xffffd671:     "аргумент1"
(gdb) x/s *(void**)($esp + 12)
0xffffd683:     "аргумент"
(gdb) x/s *(void**)($esp + 16)
0xffffd694:     "2"
(gdb) x/s *(void**)($esp + 20)
0xffffd696:     "аргумент 3"
(gdb) x/s *(void**)($esp + 24)
0x0:    <error: Cannot access memory at address 0x0>
(gdb)
```

Рис. 8 Копируем файл и отлаживаем : в стеке всё как в прошлой лабораторной говорилось

Выполнение заданий для самостоятельной работы

Скопировал из прошлой лабораторной программу и дописал её до нужных требований (Рис. [-@fig:009])

```
%include 'in_out.asm'

SECTION .data
msg db "Результат: ",0

SECTION .text
global _start


_start:

pop ecx ; Извлекаем из стека в `ecx` количество аргументов (первое значение в стеке)
pop edx ; Извлекаем из стека в `edx` имя программы (второе значение в стеке)
sub ecx,1 ; Уменьшаем `ecx` на 1 (количество аргументов без названия программы)
mov esi, 0 ; Используем `esi` для хранения промежуточных сумм

next:

cmp ecx,0h ; проверяем, есть ли еще аргументы
jz _end ; если аргументов нет выходим из цикла (переход на метку `_end`)
pop eax ; иначе извлекаем следующий аргумент из стека
call atoi ; преобразуем символ в число

call _func
add esi,eax ; добавляем к промежуточной сумме след. аргумент `esi=esi+eax`

loop next ; переход к обработке следующего аргумента


_end:

mov eax, msg ; вывод сообщения "Результат: "
call sprint
mov eax, esi ; записываем сумму в регистр `eax`
call iprintLF ; печать результата
call quit ; завершение программы


_func:
mov ebx,8
mul ebx
sub eax,3
ret ; выход из подпрограммы
~
~
~
~
~
~
"lab09-sm19-1.asm" 43L, 1524B                                 25,114-74     All
```

Рис. 9 Скопировал из прошлой лабораторной программу и дописал её до нужных требований

Правда работает (Рис. [-@fig:010])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf lab09-sm19-1.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-sm19-1 lab09-sm19-1.o
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ./lab09-sm19-1 1 2 1
Результат: 23
```

Рис. 10 Правда работает

Ввёл новый листинг (Рис. [-@fig:011])

```
%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 ebx,eax
mov ecx,4
mul ecx
add ebx,5
mov edi,ebx
; ---- Вывод результата на экран
mov eax,div
call sprint
mov eax,edi
call iprintLF
call quit
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                    21,1          All
```

Рис. 11 Ввёл новый листинг

Работает не как изначально задумано (Рис. [-@fig:012])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf lab09-sm19-2.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-sm19-2 lab09-sm19-2.o
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ./lab09-sm19-2
Результат: 10
```

Рис. 12 Работает не как изначально задумано

запускаем отладчик (Рис. [-@fig:013])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf -g -l lab09-sm19-2.lst lab09-sm19-2.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-sm19-2 lab09-sm19-2.o
tbshirinkin@Shiza:~/work/arch-pc/lab09$ gdb lab09-sm19-2
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-sm19-2...
(gdb) layout asm
```

Рис. 13 запускаем отладчик

Тут не в то регистр умножился (Рис. [-@fig:014])

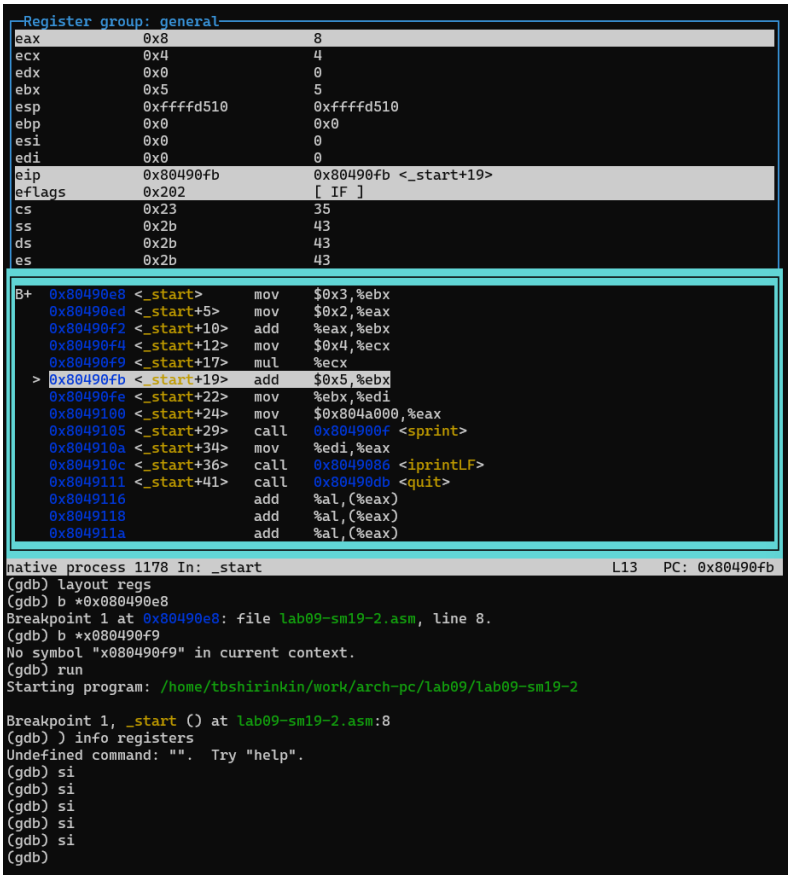

Рис. 14 Тут не в то регистр умножился

А вот откуда в выводе 10 (Рис. [-@fig:015])

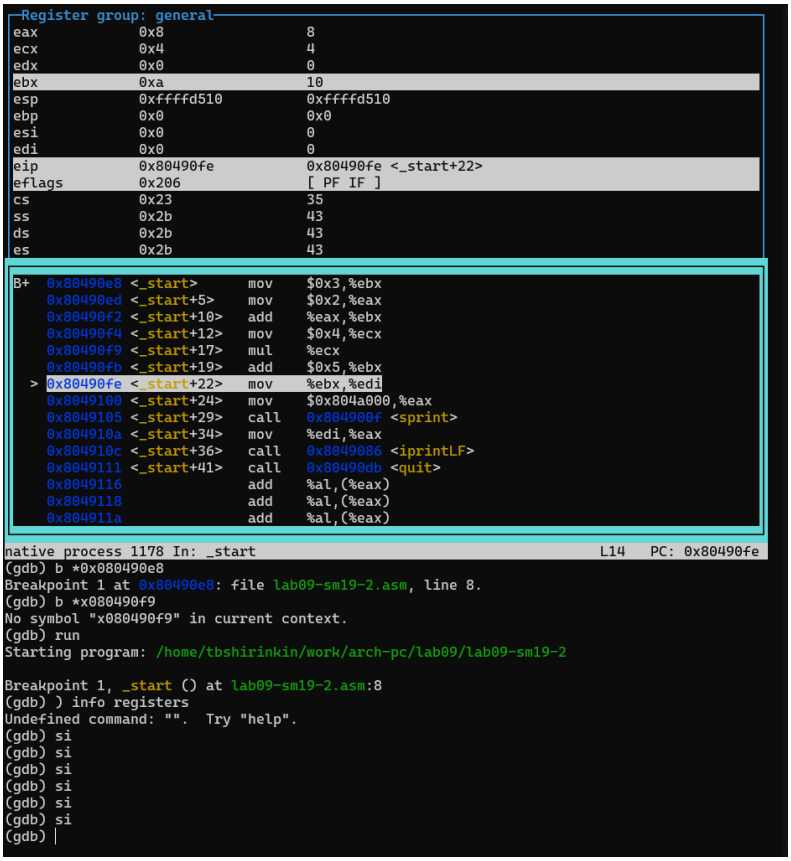

Рис. 15 А вот откуда в выводе 10

Исправил (Рис. [-@fig:016])

```
%include 'in_out.asm'
SECTION .data
div: DB 'Результат: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения (3+2)*4+5
mov eax,3
mov ebx,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
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                    20,9          All
```

Рис. 16 Исправил

Работает (Рис. [-@fig:017])

```
tbshirinkin@Shiza:~/work/arch-pc/lab09$ nasm -f elf lab09-sm19-2.asm
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ld -m elf_i386 -o lab09-sm19-2 lab09-sm19-2.o
tbshirinkin@Shiza:~/work/arch-pc/lab09$ ./lab09-sm19-2
Результат: 25
```

Выводы

Цель достигнута: приобретены навыки программирования с использованием подпрограмм, совершено знакомство с отладкой при помощи gdb.