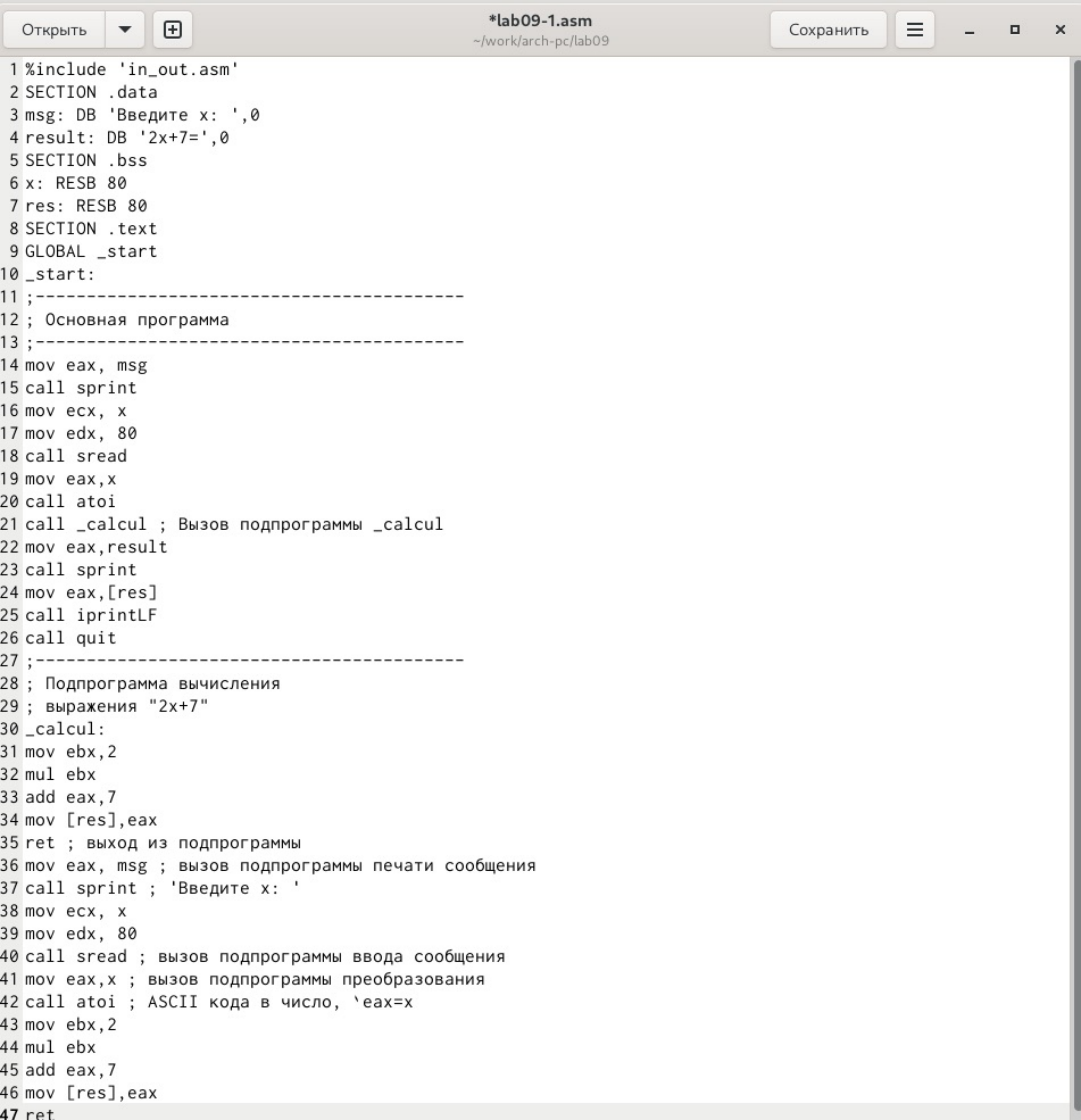
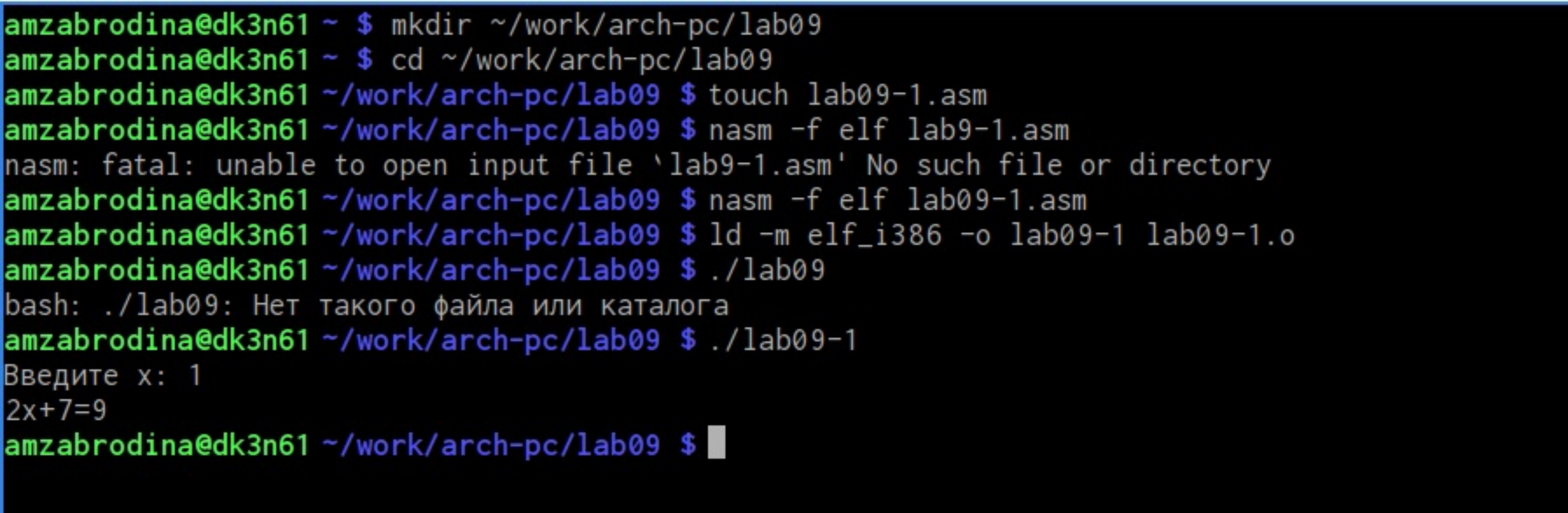
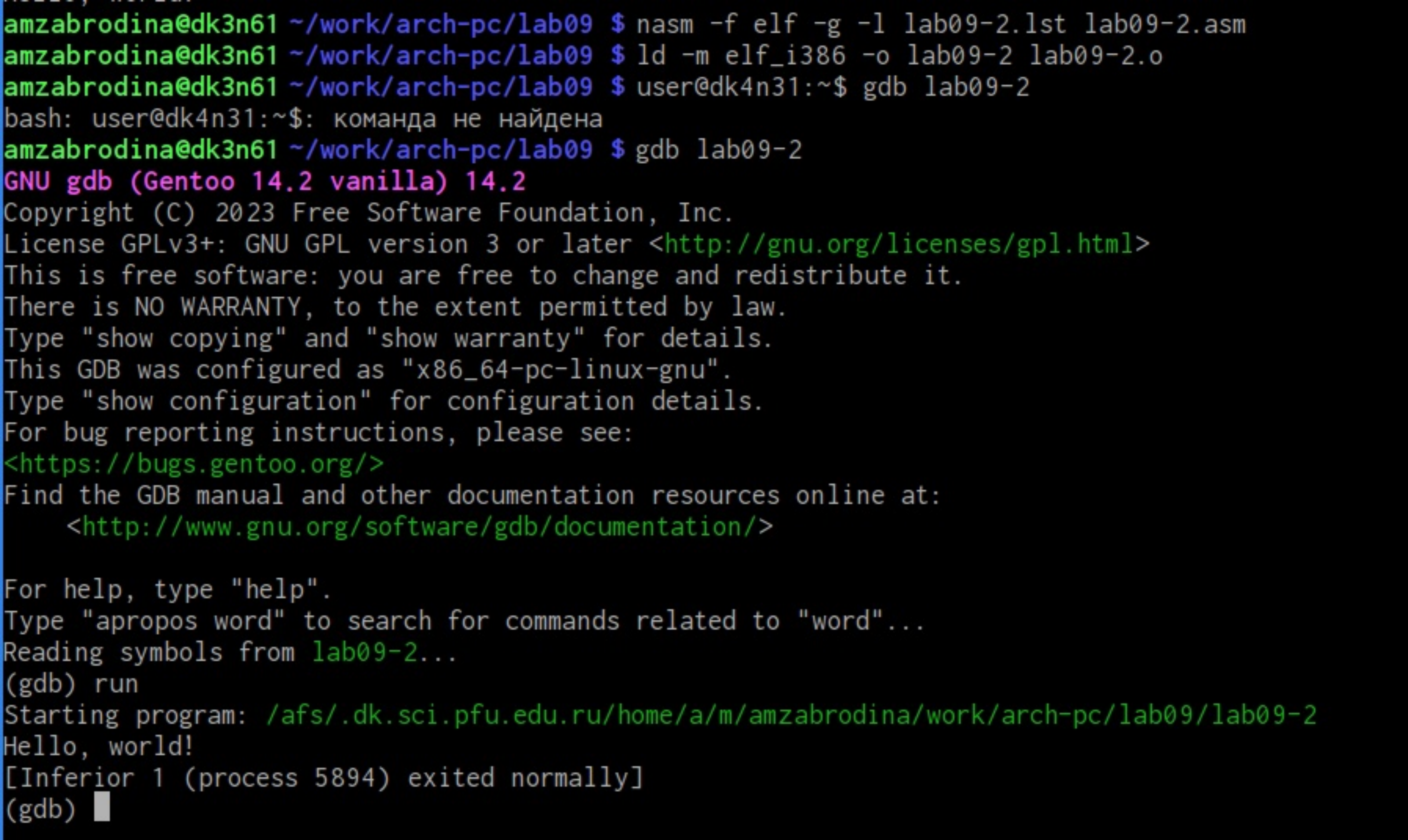


```
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите x: ',0
result: DB '2x+7=',0
SECTION .bss
x: RESB 80
res: RESB 80
SECTION .text
GLOBAL _start
_start:
;-------------------------------------------
; Основная программа
;-------------------------------------------
mov eax, msg
call sprint
mov ecx, x
mov edx, 80
call sread
mov eax,x
call atoi
call _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
mov eax, msg ; вызов подпрограммы печати сообщения
call sprint ; 'Введите x: '
mov ecx, x
mov edx, 80
call sread ; вызов подпрограммы ввода сообщения
mov eax,x ; вызов подпрограммы преобразования
call atoi ; ASCII кода в число, `eax=x
mov ebx,2
mul ebx
add eax,7
mov [res],eax
ret
```



```
amzabrodina@dk3n61 ~ $ mkdir ~/work/arch-pc/lab09
amzabrodina@dk3n61 ~ $ cd ~/work/arch-pc/lab09
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ touch lab09-1.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf lab9-1.asm
nasm: fatal: unable to open input file `lab9-1.asm' No such file or directory
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf lab09-1.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-1 lab09-1.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09
bash: ./lab09: Нет такого файла или каталога
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09-1
Введите x: 1
2x+7=9
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```



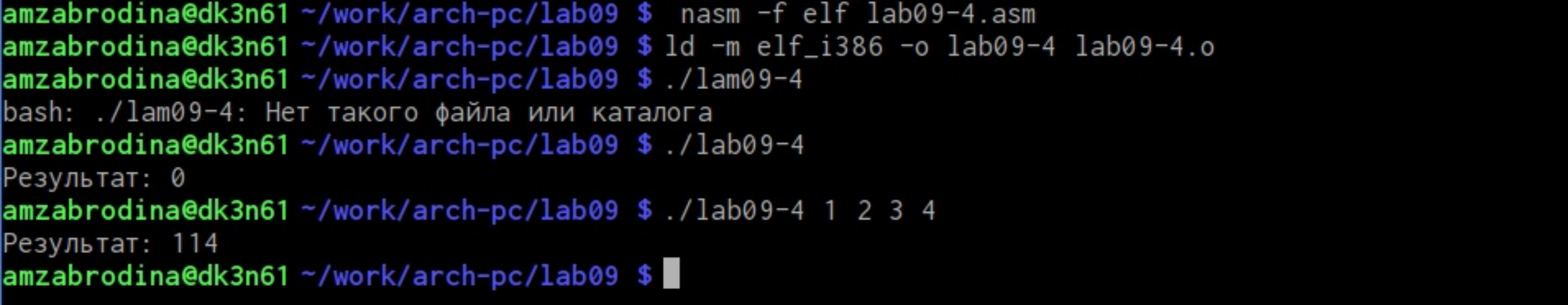
```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf -g -l lab09-2.lst lab09-2.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-2 lab09-2.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ user@dk4n31:~$ gdb lab09-2
bash: user@dk4n31:~$: команда не найдена
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ gdb lab09-2
GNU gdb (Gentoo 14.2 vanilla) 14.2
Copyright (C) 2023 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-pc-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://bugs.gentoo.org/>
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) run
Starting program: /afs/.dk.sci.pfu.edu.ru/home/a/m/amzabrodina/work/arch-pc/lab09/lab09-2
Hello, world!
[Inferior 1 (process 5894) exited normally]
(gdb) 
```

```
(gdb) layout regs
(gdb) info breakpoints
Num     Type           Disp Enb Address    What
1       breakpoint     keep y   0x08049000 lab09-2.asm:9
        breakpoint already hit 1 time
(gdb) break *0x8049000
Note: breakpoint 1 also set at pc 0x8049000.
Breakpoint 2 at 0x8049000: file lab09-2.asm, line 9.
(gdb) 
```

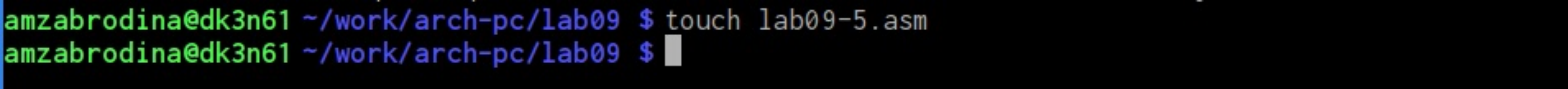
```
native process 7698 In: _start                                                                                     L9    PC: 0x8049000
esi            0x0                 0
edi            0x0                 0
eip            0x8049000           0x8049000 <_start>
eflags         0x202               [ IF ]
cs             0x23                35
ss             0x2b                43
ds             0x2b                43
es             0x2b                43
fs             0x0                 0
--Type <RET> for more, q to quit, c to continue without paging--q
Quit
(gdb) x/1sb &msg1
0x804a000 <msg1>:       "Hello, "
(gdb) x/1sb 0x804a008
0x804a008 <msg2>:       "world!\n\034"
(gdb) 
```

Файл Действия Правка Вид Справка
amzabrodina@dk3n61 - lab09
amzabrodina@dk3n61 - lab09
Архитектура ЭВМ - Ð_Ð°Ð±Ð¾...
lab09
lab09-2.asm (~/work/arch-pc/lab...
amzabrodina@dk3n61 - lab09

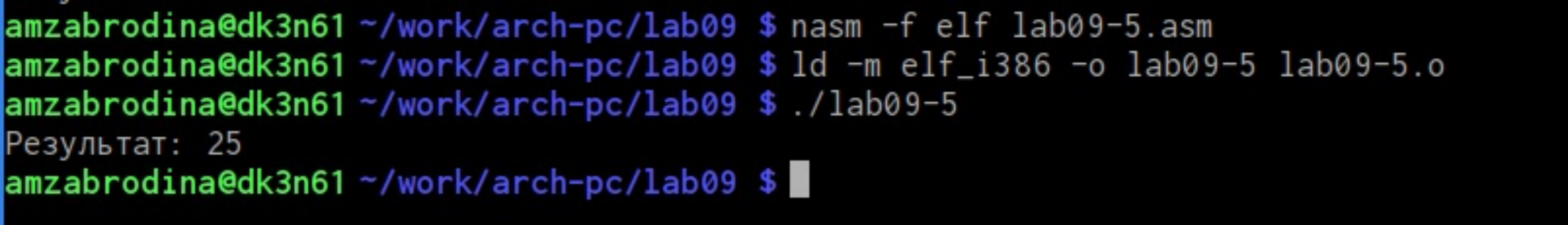
```
                              [ Register Values Unavailable ]

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
   0x8049020 <_start+32>    mov    ecx,0x804a008
   0x8049025 <_start+37>    mov    edx,0x7
   0x804902a <_start+42>    int    0x80
   0x804902c <_start+44>    mov    eax,0x1
   0x8049031 <_start+49>    mov    ebx,0x0
   0x8049036 <_start+54>    int    0x80
   0x8049038                 add    BYTE PTR [eax],al
native process 7698 In: _start                                                           L9    PC: 0x8049000
eax            0x0                 0
ecx            0x0                 0
edx            0x0                 0
ebx            0x0                 0
esp            0xffffc3c0          0xffffc3c0
ebp            0x0                 0x0
esi            0x0                 0
edi            0x0                 0
eip            0x8049000           0x8049000 <_start>
eflags         0x202               [ IF ]
cs             0x23                35
ss             0x2b                43
ds             0x2b                43
es             0x2b                43
fs             0x0                 0
--Type <RET> for more, q to quit, c to continue without paging--
```



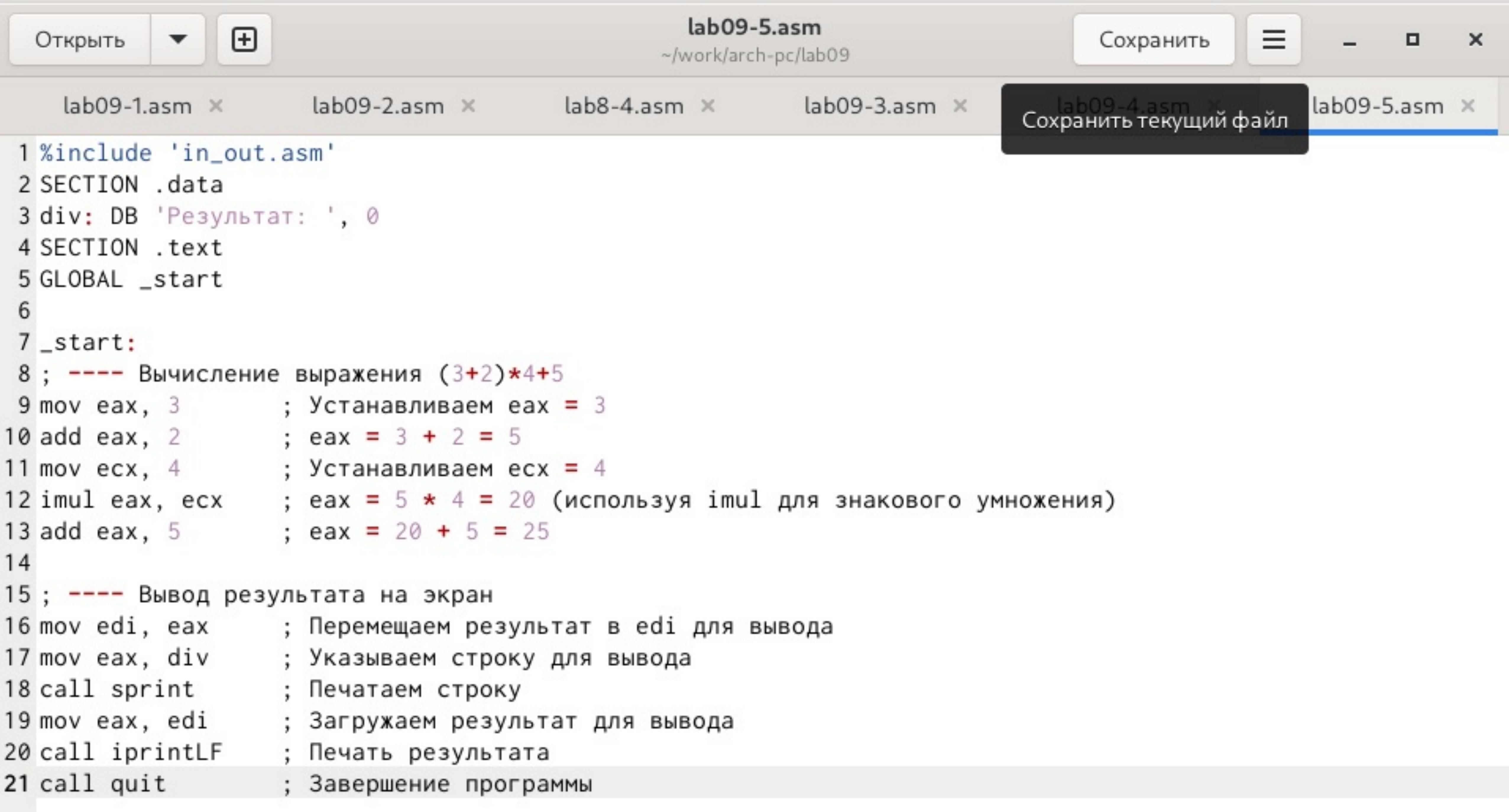
```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $  nasm -f elf lab09-4.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-4 lab09-4.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lam09-4
bash: ./lam09-4: Нет такого файла или каталога
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09-4
Результат: 0
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09-4 1 2 3 4
Результат: 114
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```



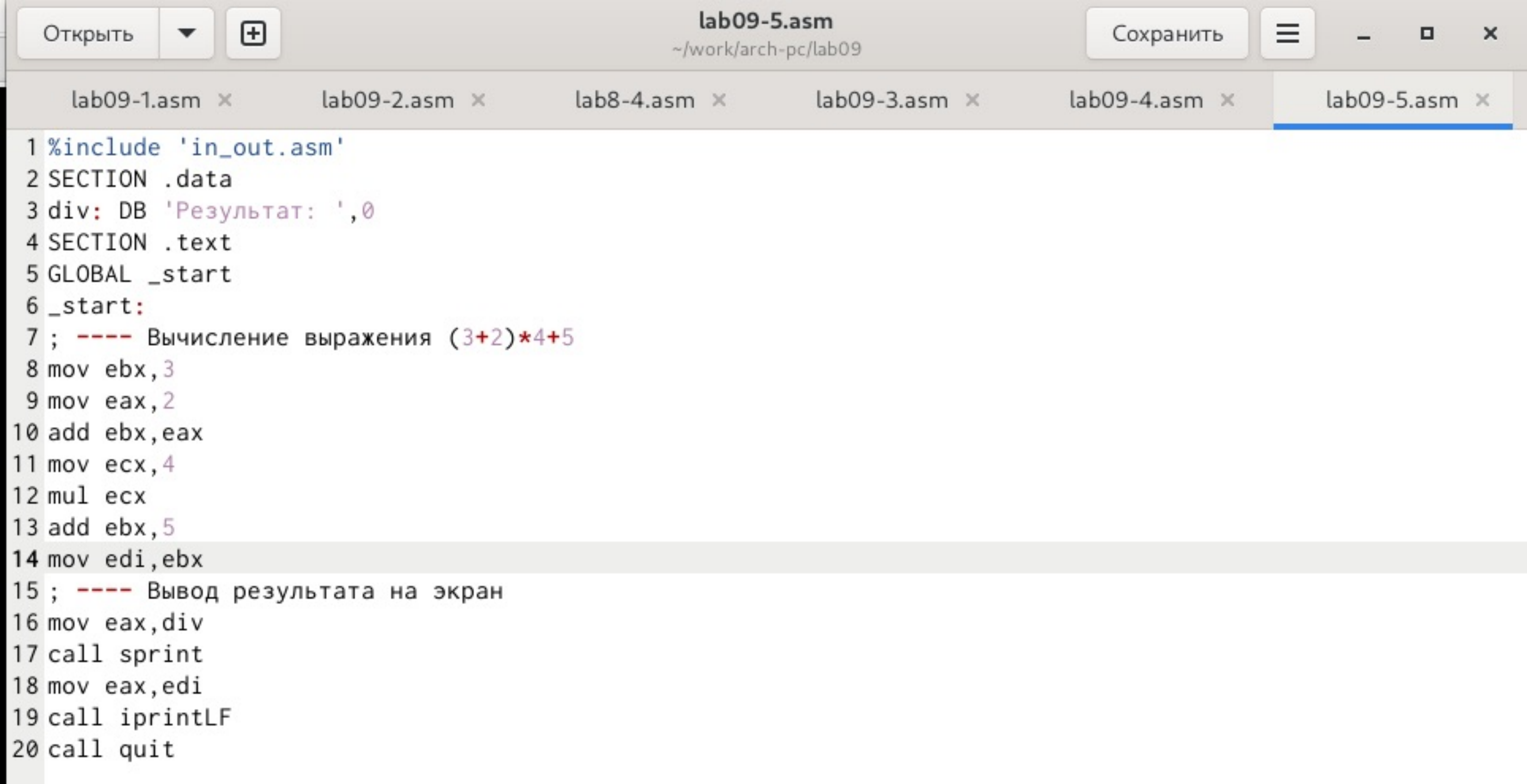
```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ touch lab09-5.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```



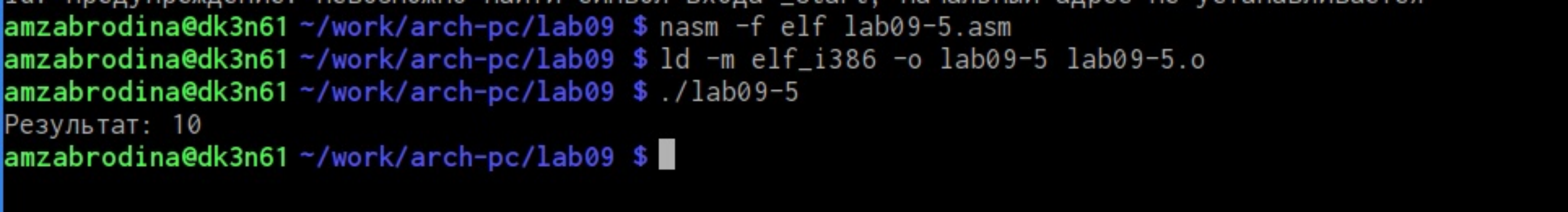
```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf lab09-5.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-5 lab09-5.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09-5
Результат: 25
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```



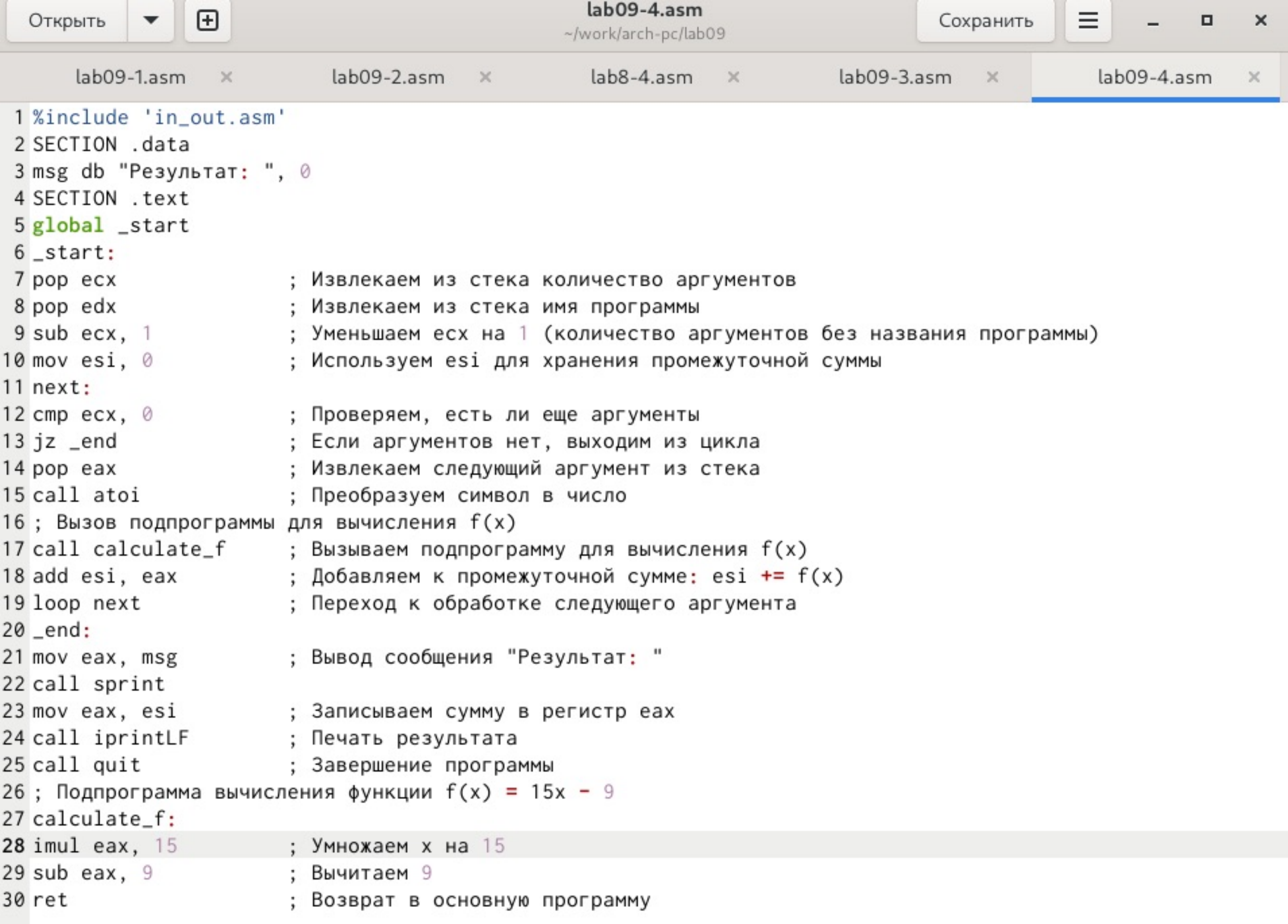
```
%include 'in_out.asm'
SECTION .data
div: DB 'Результат: ', 0
SECTION .text
GLOBAL _start

_start:
; ---- Вычисление выражения (3+2)*4+5
mov eax, 3       ; Устанавливаем eax = 3
add eax, 2       ; eax = 3 + 2 = 5
mov ecx, 4       ; Устанавливаем ecx = 4
imul eax, ecx    ; eax = 5 * 4 = 20 (используя imul для знакового умножения)
add eax, 5       ; eax = 20 + 5 = 25

; ---- Вывод результата на экран
mov edi, eax     ; Перемещаем результат в edi для вывода
mov eax, div     ; Указываем строку для вывода
call sprint      ; Печатаем строку
mov eax, edi     ; Загружаем результат для вывода
call iprintLF    ; Печать результата
call quit        ; Завершение программы
```



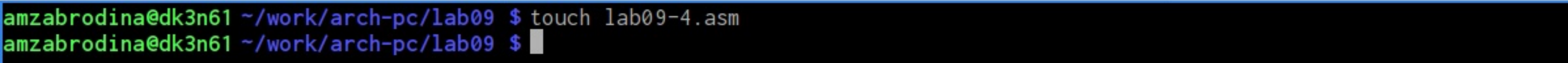
```
%include 'in_out.asm'
SECTION .data
div: DB 'Результат: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения (3+2)*4+5
mov ebx,3
mov eax,2
add ebx,eax
mov ecx,4
mul ecx
add ebx,5
mov edi,ebx
; ---- Вывод результата на экран
mov eax,div
call sprint
mov eax,edi
call iprintLF
call quit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



```
ld: предупреждение: невозможно найти символ входа _start; начальный адрес не устанавливается
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf lab09-5.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-5 lab09-5.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09-5
Результат: 10
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```



```
%include 'in_out.asm'
SECTION .data
msg db "Результат: ", 0
SECTION .text
global _start
_start:
pop ecx              ; Извлекаем из стека количество аргументов
pop edx              ; Извлекаем из стека имя программы
sub ecx, 1           ; Уменьшаем ecx на 1 (количество аргументов без названия программы)
mov esi, 0           ; Используем esi для хранения промежуточной суммы
next:
cmp ecx, 0           ; Проверяем, есть ли еще аргументы
jz _end              ; Если аргументов нет, выходим из цикла
pop eax              ; Извлекаем следующий аргумент из стека
call atoi            ; Преобразуем символ в число
; Вызов подпрограммы для вычисления f(x)
call calculate_f     ; Вызываем подпрограмму для вычисления f(x)
add esi, eax         ; Добавляем к промежуточной сумме: esi += f(x)
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
; Подпрограмма вычисления функции f(x) = 15x - 9
calculate_f:
imul eax, 15         ; Умножаем x на 15
sub eax, 9           ; Вычитаем 9
ret                  ; Возврат в основную программу
```

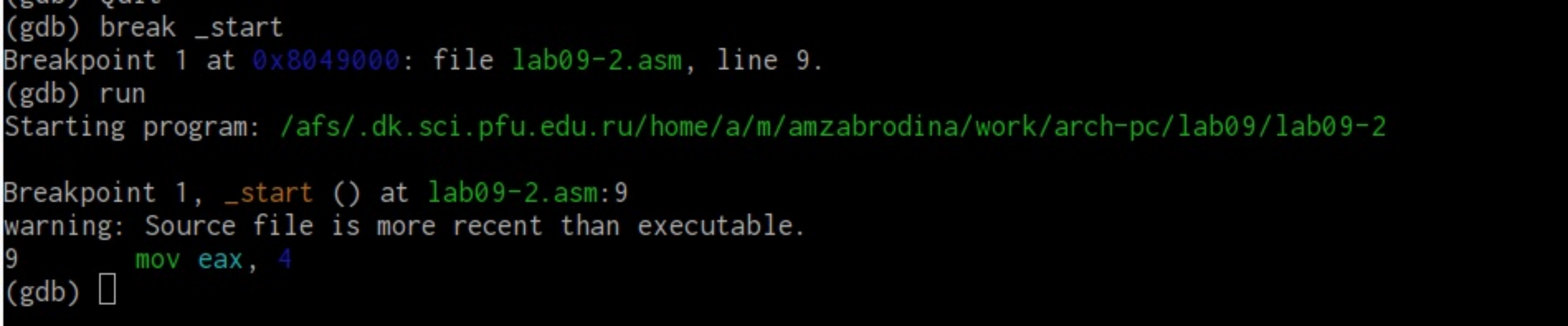


```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ touch lab09-4.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```

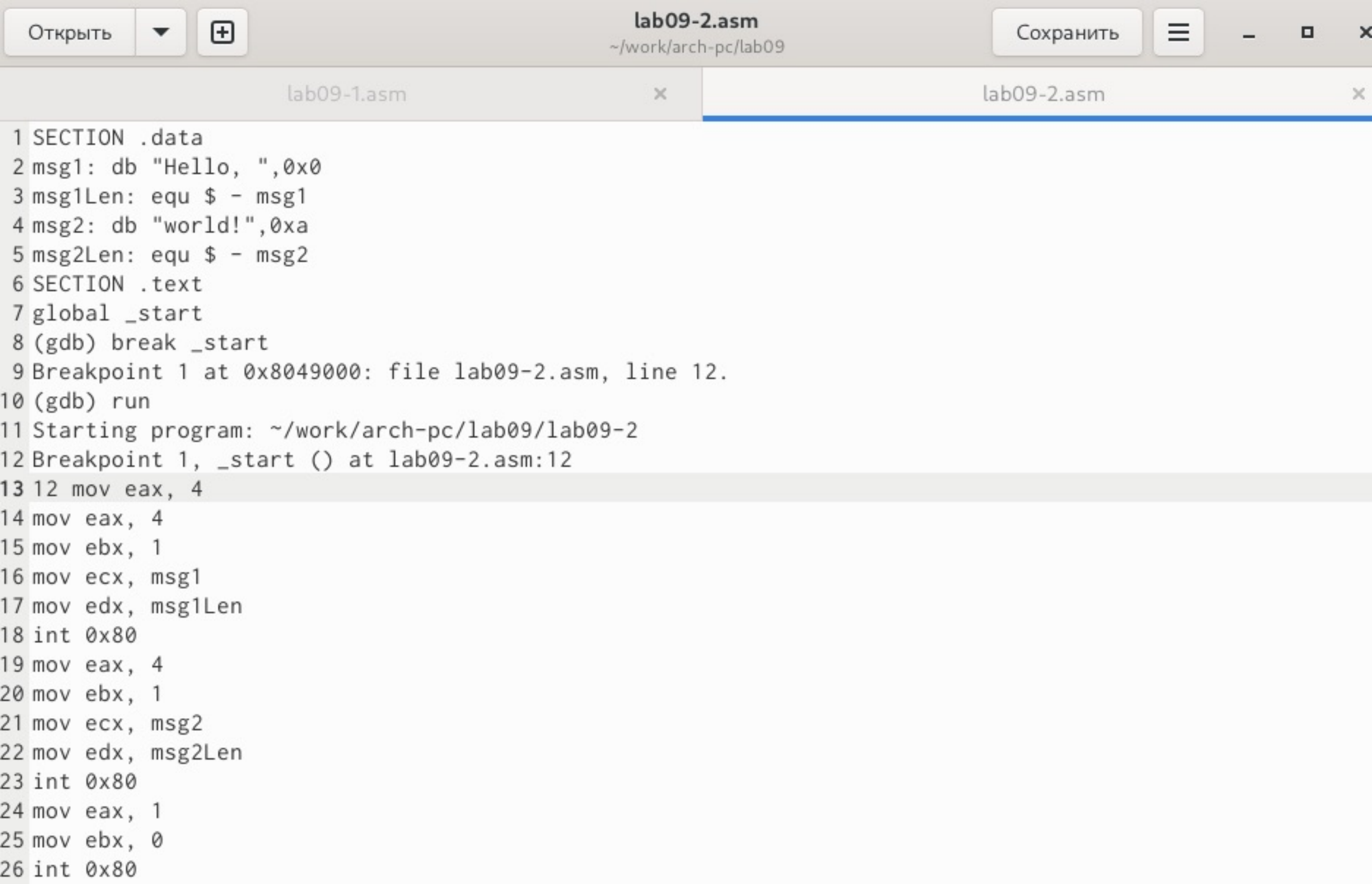
```
(gdb) x/x $esp
0xffffc370:     0x00000005
(gdb) x/s *(void**)($esp + 4)
0xffffc5ef:     "/afs/.dk.sci.pfu.edu.ru/home/a/m/amzabrodina/work/arch-pc/lab09/lab09-3"
(gdb) (gdb) x/s *(void**)($esp + 8)
Undefined command: "".  Try "help".
(gdb) x/s *(void**)($esp + 8)
0xffffc637:     "аргумент1"
(gdb) x/s *(void**)($esp + 12)
0xffffc649:     "аргумент"
(gdb) x/s *(void**)($esp + 16)
0xffffc65a:     "2"
(gdb) x/s *(void**)($esp + 20)
0xffffc65c:     "аргумент 3"
(gdb) x/s *(void**)($esp + 24)
0x0:    <error: Cannot access memory at address 0x0>
(gdb) 
```

```
GNU gdb (Gentoo 14.2 vanilla) 14.2
Copyright (C) 2023 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-pc-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://bugs.gentoo.org/>
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: /afs/.dk.sci.pfu.edu.ru/home/a/m/amzabrodina/work/arch-pc/lab09/lab09-3 аргумент1 аргумент 2 аргумент\ 3

Breakpoint 1, _start () at lab09-3.asm:5
5           pop ecx ;
(gdb) 
```

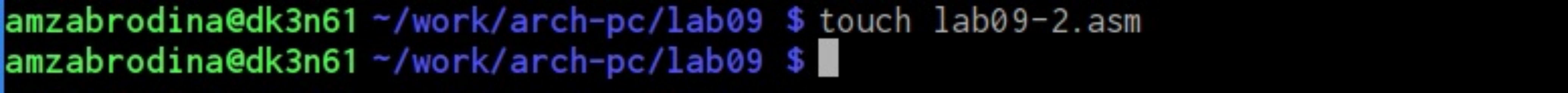
```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ touch lab09-3.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ cp ~/work/arch-pc/lab08/lab8-2.asm ~/work/arch-pc/lab09/lab09-3.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf -g -l lab09-3.lst lab09-3.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-3 lab09-3.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ gdb --args lab09-3 аргумент1 аргумент 2 'аргумент 3'
GNU gdb (Gentoo 14.2 vanilla) 14.2
Copyright (C) 2023 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-pc-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://bugs.gentoo.org/>.
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) 
```



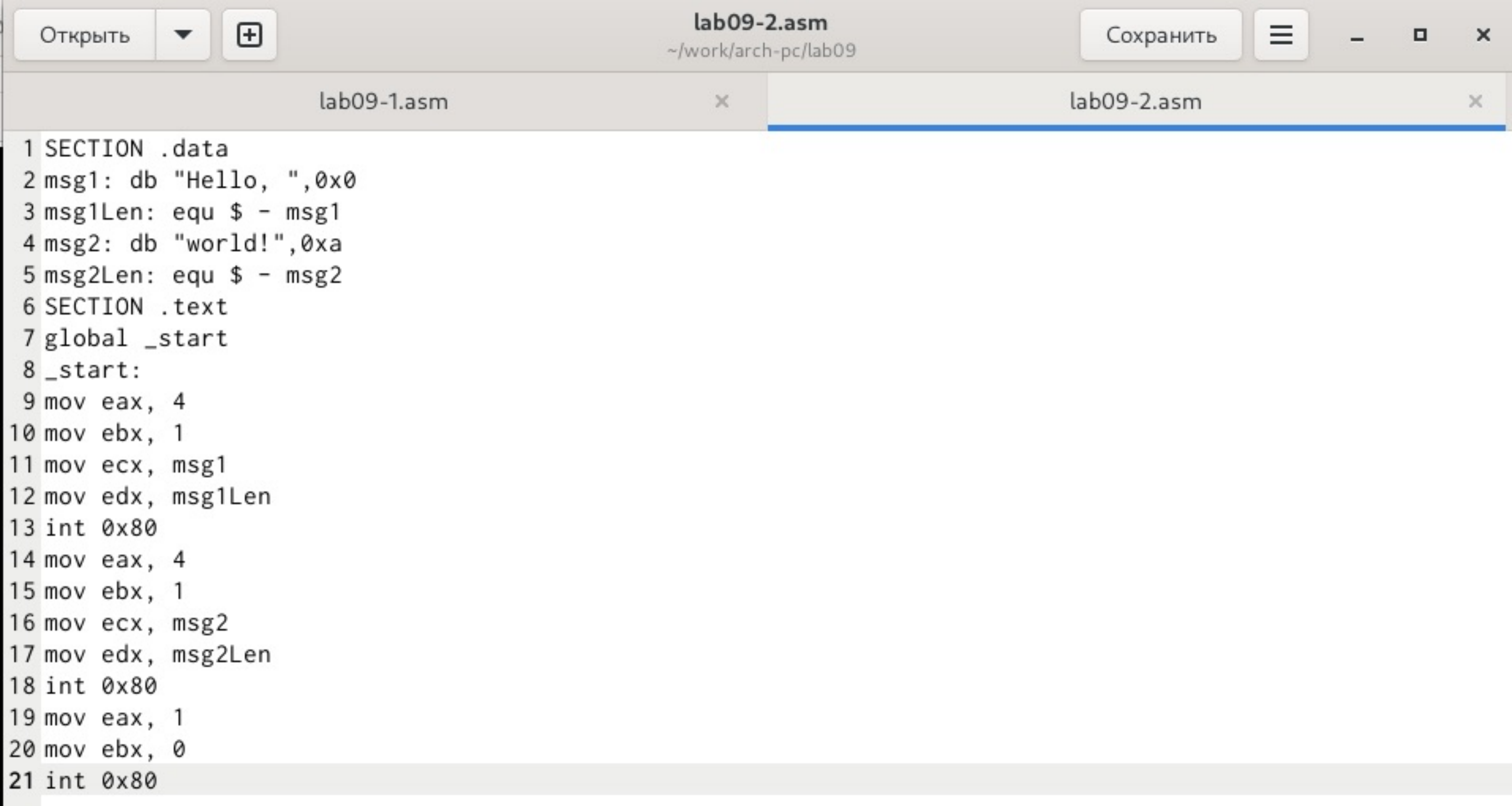
```
(gdb) quit
(gdb) break _start
Breakpoint 1 at 0x8049000: file lab09-2.asm, line 9.
(gdb) run
Starting program: /afs/.dk.sci.pfu.edu.ru/home/a/m/amzabrodina/work/arch-pc/lab09/lab09-2

Breakpoint 1, _start () at lab09-2.asm:9
warning: Source file is more recent than executable.
9           mov eax, 4
(gdb) 
```



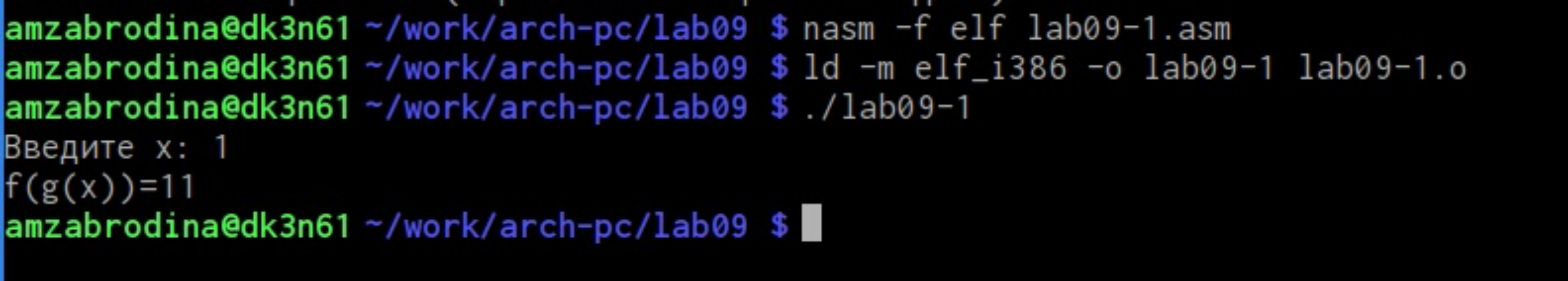
```
SECTION .data
msg1: db "Hello, ",0x0
msg1Len: equ $ - msg1
msg2: db "world!",0xa
msg2Len: equ $ - msg2
SECTION .text
global _start
(gdb) break _start
Breakpoint 1 at 0x8049000: file lab09-2.asm, line 12.
(gdb) run
Starting program: ~/work/arch-pc/lab09/lab09-2
Breakpoint 1, _start () at lab09-2.asm:12
12 mov eax, 4
mov eax, 4
mov ebx, 1
mov ecx, msg1
mov edx, msg1Len
int 0x80
mov eax, 4
mov ebx, 1
mov ecx, msg2
mov edx, msg2Len
int 0x80
mov eax, 1
mov ebx, 0
int 0x80
```



```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ touch lab09-2.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```

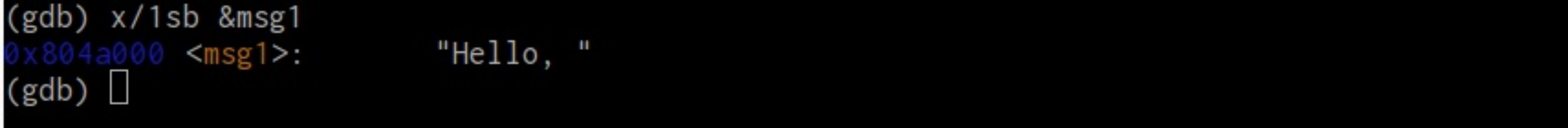


```
SECTION .data
msg1: db "Hello, ",0x0
msg1Len: equ $ - msg1
msg2: db "world!",0xa
msg2Len: equ $ - msg2
SECTION .text
global _start
_start:
mov eax, 4
mov ebx, 1
mov ecx, msg1
mov edx, msg1Len
int 0x80
mov eax, 4
mov ebx, 1
mov ecx, msg2
mov edx, msg2Len
int 0x80
mov eax, 1
mov ebx, 0
int 0x80
```

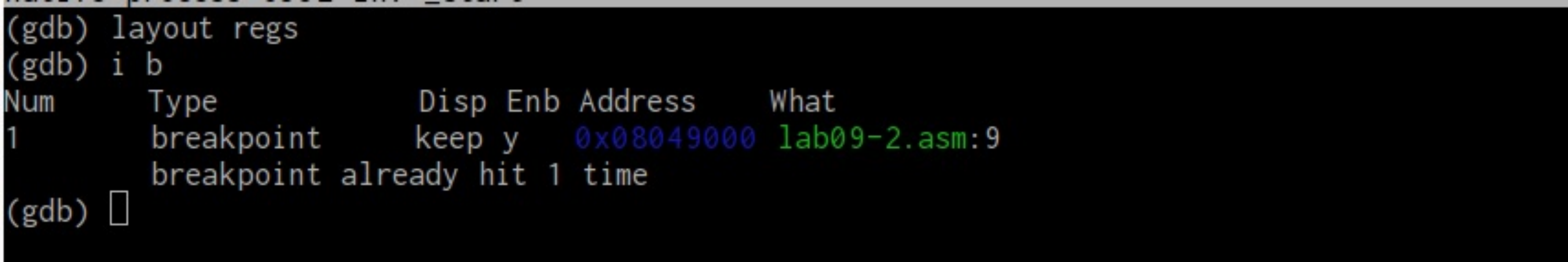


```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf lab09-1.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-1 lab09-1.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ./lab09-1
Введите x: 1
f(g(x))=11
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```

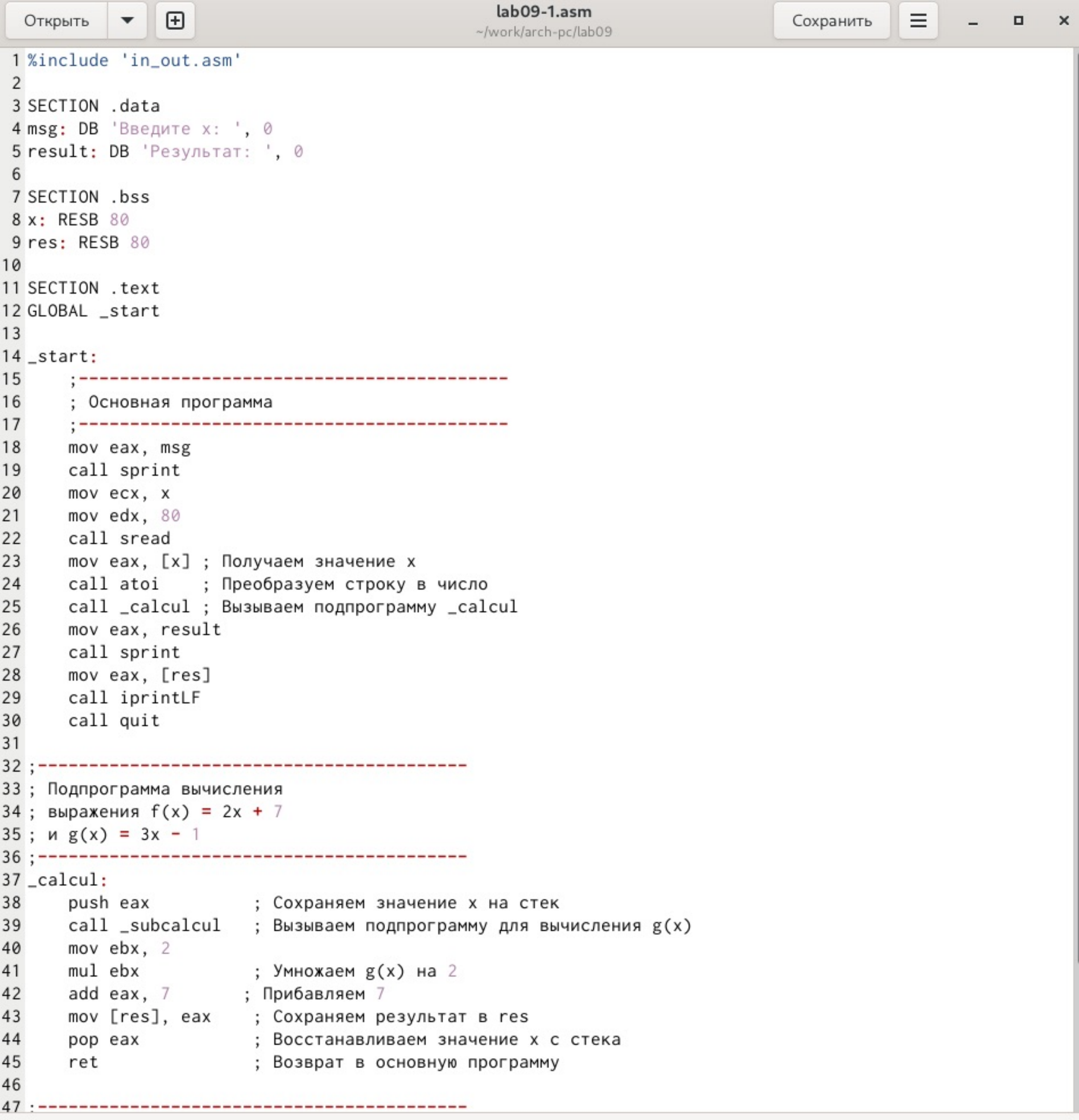
```
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ touch lab09-3.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ cp ~/work/arch-pc/lab08/lab8-2.asm ~/work/arch-pc/lab09/lab09-3.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ nasm -f elf -g -l lab09-3.lst lab09-3.asm
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab09-3 lab09-3.o
amzabrodina@dk3n61 ~/work/arch-pc/lab09 $ 
```



```
(gdb) x/1sb &msg1
0x804a000 <msg1>:       "Hello, "
(gdb) 
```



```
(gdb) layout regs
(gdb) i b
Num     Type           Disp Enb Address    What
1       breakpoint     keep y   0x08049000 lab09-2.asm:9
        breakpoint already hit 1 time
(gdb) 
```



```
%include 'in_out.asm'

SECTION .data
msg: DB 'Введите x: ', 0
result: DB 'Результат: ', 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] ; Получаем значение x
    call atoi    ; Преобразуем строку в число
    call _calcul ; Вызываем подпрограмму _calcul
    mov eax, result
    call sprint
    mov eax, [res]
    call iprintLF
    call quit

;------------------------------------------
; Подпрограмма вычисления
; выражения f(x) = 2x + 7
; и g(x) = 3x - 1
;------------------------------------------
_calcul:
    push eax          ; Сохраняем значение x на стек
    call _subcalcul   ; Вызываем подпрограмму для вычисления g(x)
    mov ebx, 2
    mul ebx           ; Умножаем g(x) на 2
    add eax, 7       ; Прибавляем 7
    mov [res], eax    ; Сохраняем результат в res
    pop eax           ; Восстанавливаем значение x с стека
    ret               ; Возврат в основную программу

;------------------------------------------
```