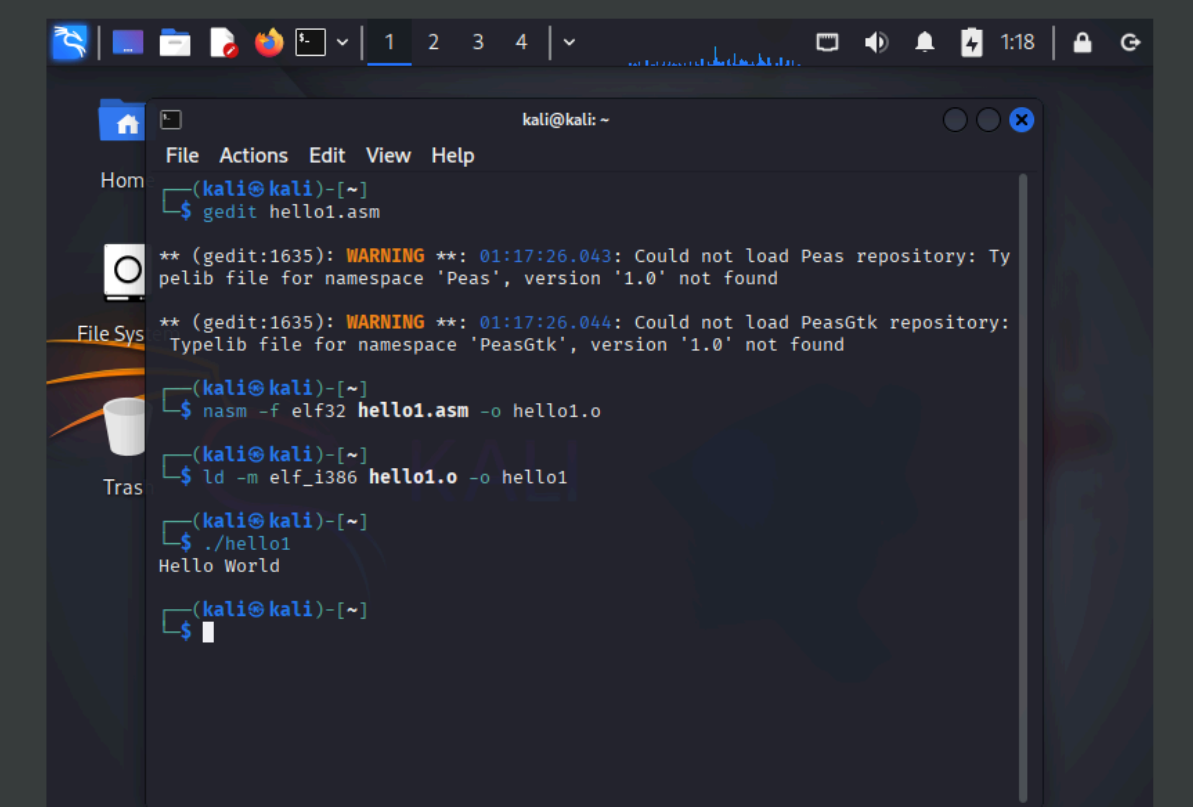```asm
section .data
msg db 'Hello World',0xa
len equ $-msg

section .text
global _start

_start:
    mov eax,4
    mov ebx,1
    mov ecx,msg
    mov edx,len
    int 0x80
    mov eax,1
    xor ebx,ebx
    int 0x80
```
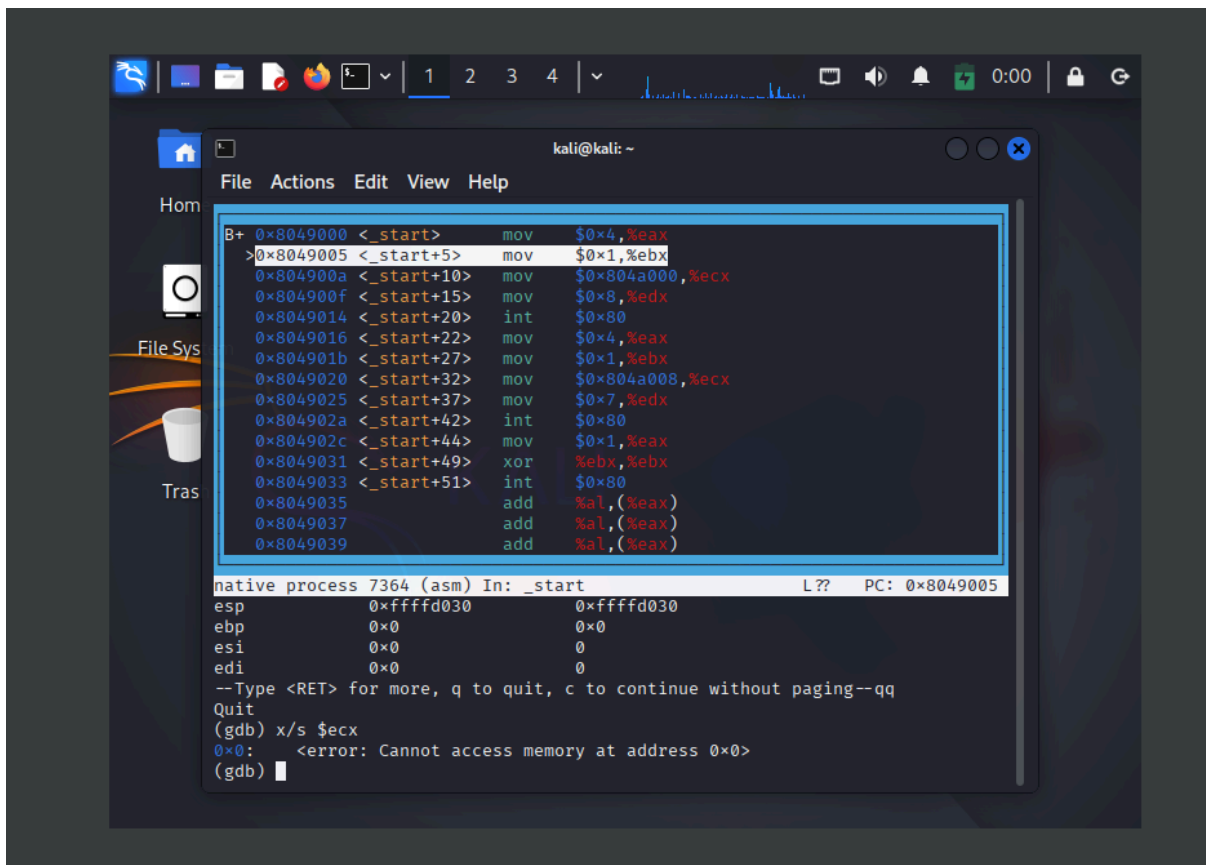


```
gdb ./hello

set disassembly-flavor intel
break _start
run
layout asm
stepi
```

; hello.asm - Prints first name and second name on different lines (32-bit NASM, Linux)

section .data
    firstName db "Atharva", 0x0A ; First name + newline
    firstLen equ $ - firstName
    secondName db "Gadgil", 0x0A ; Second name + newline
    secondLen equ $ - secondName

section .text
    global _start

_start:
    ; Print first name
    mov eax, 4   ; sys_write
    mov ebx, 1   ; stdout
    mov ecx, firstName ; message address
    mov edx, firstLen   ; message length
    int 0x80

    ; Print second name
    mov eax, 4
    mov ebx, 1
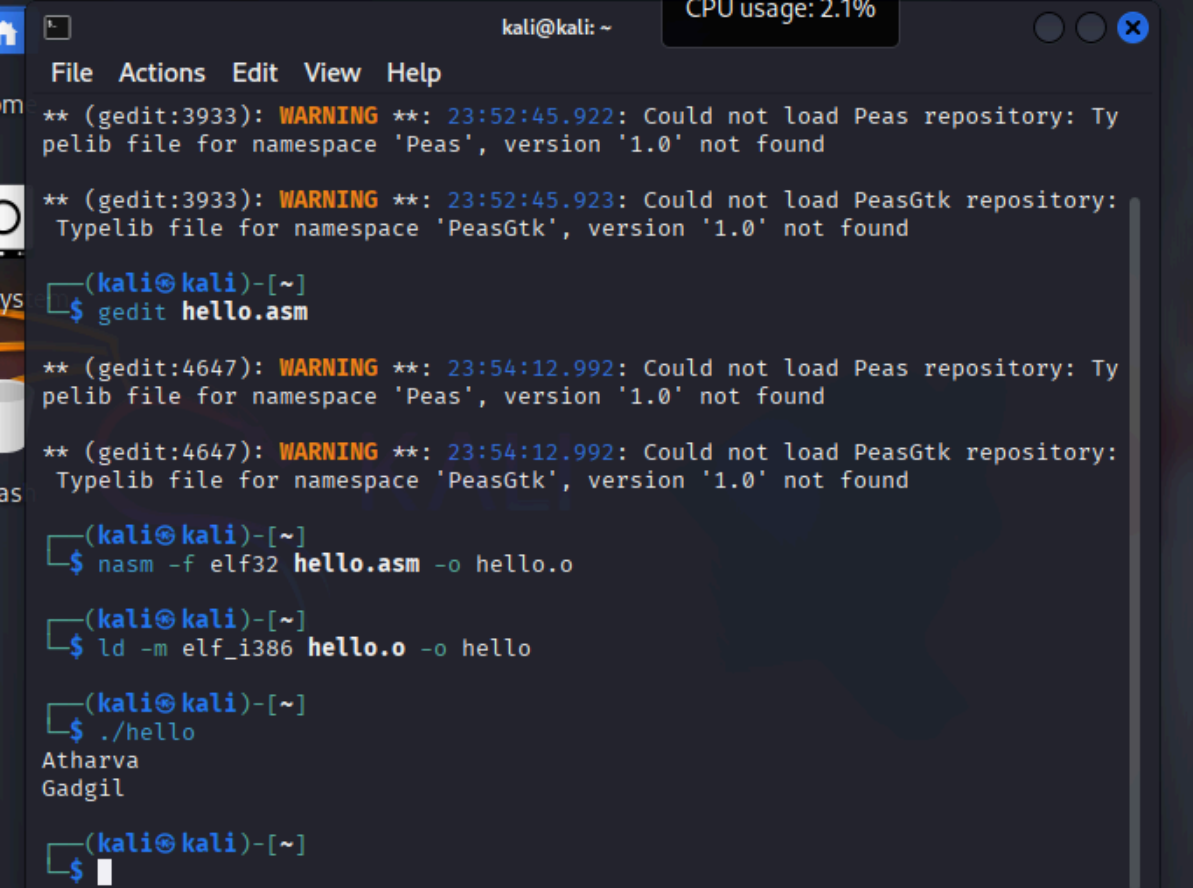   mov ecx, secondName
    mov edx, secondLen

```
int 0x80

; Exit program
mov eax, 1        ; sys_exit
xor ebx, ebx      ; exit code 0
int 0x80
```