

Oblig 3

onsdag 3. oktober 2018 09:10

Oppgave 1:

Hello.asm:

Sondre Lindaas Gjesdal, Espen Haugland
Mæland og Herborg Irgens Sjø

```
helloasm (~/.dat103/oblig3) - gedit
Open [icon]

; Program som skriver ut Hello World

; Konstantar
    cr equ 13      ; Vognretur
    lf equ 10      ; Linjeskift

section .data      ; Datasegment
    melding db 'Hello World!', cr, lf
    lengde equ $ - melding

section .text      ; Kodesegment
global _start
_start:
    mov edx, lengde
    mov ecx, melding
    mov ebx, 1
    mov eax, 4      ; sys_write
    int 80h
    mov ebx, 0
    mov eax, 1      ; sys_exit
    int 80h
```

Køyring av programmet i terminalen:

```
heborg@heborg-VirtualBox:~/dat103/oblig3$ ./hello
Hello World!
```

Opsjonar i gdb:

layout regs:

Opner eit vindauge som viser kva som blir registrert i kvar linja av koden. Dersom det eksisterande vindauget er assembly, så blir registrering vindauge opna.

b:

Setter eit bruddpunkt på ein bestemt stad, i dette tilfelle så blir det plassert i start.

r:

Startar debugg programmet, kan brukast meir spesifikk, men det gjer me ikkje her.

si:

Gjer kun ein instruksjon, det vil seie at den kun går gjennom ei linja av koden.

s:

Gjer eit steg i koden til den treff ei linja som er annleis.

Oppgave 2:

Oppg2.asm:

Det er kun gjort endring på metoden skrivsiffer frå koden me fekk, difor tek eg kun med eit skjermbileta av den metoden.

```

; -----
skrivsiffer:
    ; Skriver ut sifferet lagret i ecx. Ingen sjekk på verdioråde.
    push eax
    push ebx
    push ecx
    push edx

    cmp ecx, 10
    jb eitsiffer
    mov ebx, ecx
    mov ecx, 1
    call skrivsiffer
    sub ebx, 10
    mov ecx, ebx
eitsiffer:

    add ecx, '0' ; converter tall til ascii.
    mov [siffer], ecx
    mov ecx, siffer
    mov edx, 1
    mov ebx, STDOUT
    mov eax, SYS_WRITE
    int 80h
    pop edx
    pop ecx
    pop ebx
    pop eax
    ret
; -----

```

Køyring av programmet i gdb:

```
@heborg-VirtualBox: ~/dat103/oblig3
eax      0x4      4
eax      0x1      1
ecx      0xb      11
edx      0x0      0
ebx      0x0      0
esp      0xffffd070 0xffffd070
cs       0x23     11c35 11c <Lokke+20>
ss       0x2b     11c43 11c <Lokke+20>
eflags   0x246    174[ PF ZF IF ] 174 <nylinje+24>
eflags   0x246    0c6[ PF ZF IF ] 0c6 <Slutt+10>
eflags   0x202    [ IF ]
gs       0x0      0

oppg2.asm
27      crlfen equ $ - crlf
28
96      section push eax ogram
96      section push eax
132     push eax

[ No Source Available ]

5      nylinje
(gdb) sNo process In:
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
eitsiffer () at oppg2.asm:90
(gdb) s
Slutt () at oppg2.asm:59
(gdb) s
(gdb) s
(gdb) s
[Inferior 1 (process 4518) exited normally]
(gdb) █
```

Køyring av programmet i terminalen:

```
@heborg-VirtualBox: ~/dat103/oblig3
heborg@heborg-VirtualBox:~/dat103/oblig3$ ./oppg2
Skriv to einsiffrige tal skilt med mellomrom.
Summen av tala må vere mindre enn 19.
8 8

16heborg@heborg-VirtualBox:~/dat103/oblig3$
heborg@heborg-VirtualBox:~/dat103/oblig3$
```

Oppgave 3:

Opppg3.asm:

```
cr equ 13 ; Vognretur
lf equ 10 ; Linjeskift
SYS_EXIT equ 1
SYS_READ equ 3
SYS_WRITE equ 4
STDIN equ 0
STDOUT equ 1
STDERR equ 2

section .bss
a resb 4
i resb 4

section .data
crlf db cr,lf
crlflen equ $ - crlf

section .text
global _start

_start:

mov [a], byte 0
mov [i], byte 0

for:
for_cond:
cmp [i], byte 20 ; sammenligne i med 20
jge for_end

cmp [i], byte 10
jge elseBody ; hopper til elseBody om a er større enn 10
inc byte [a] ; a++
jmp ifEnd

elseBody:
dec byte [a] ; a--

ifEnd:

for_update:
inc byte [i]
jmp for

for_end:

; Printer og exit
add byte [a], '0'
mov ecx, a
mov edx, 1
mov ebx, STDOUT
mov eax, SYS_WRITE
int 80h

call nyLinje

mov eax, 1
mov ebx, 0
int 80h

nyLinje:
push eax
push ebx
push ecx
push edx
mov edx, crlf
mov ecx, crlf
mov ebx, STDOUT
mov eax, SYS_WRITE
int 80h
pop edx
pop ecx
pop ebx
pop eax
ret
```

Køyring av programmet i terminalen:

```
heborg@heborg-VirtualBox:~/dat103/DAT103_Oblig3_Done/DAT103/Oblig3$ ./Oppgave3
0
heborg@heborg-VirtualBox:~/dat103/DAT103_Oblig3_Done/DAT103/Oblig3$
```