TUGAS 4
BAHASA PEMROGRAMAN ASSEMBLY

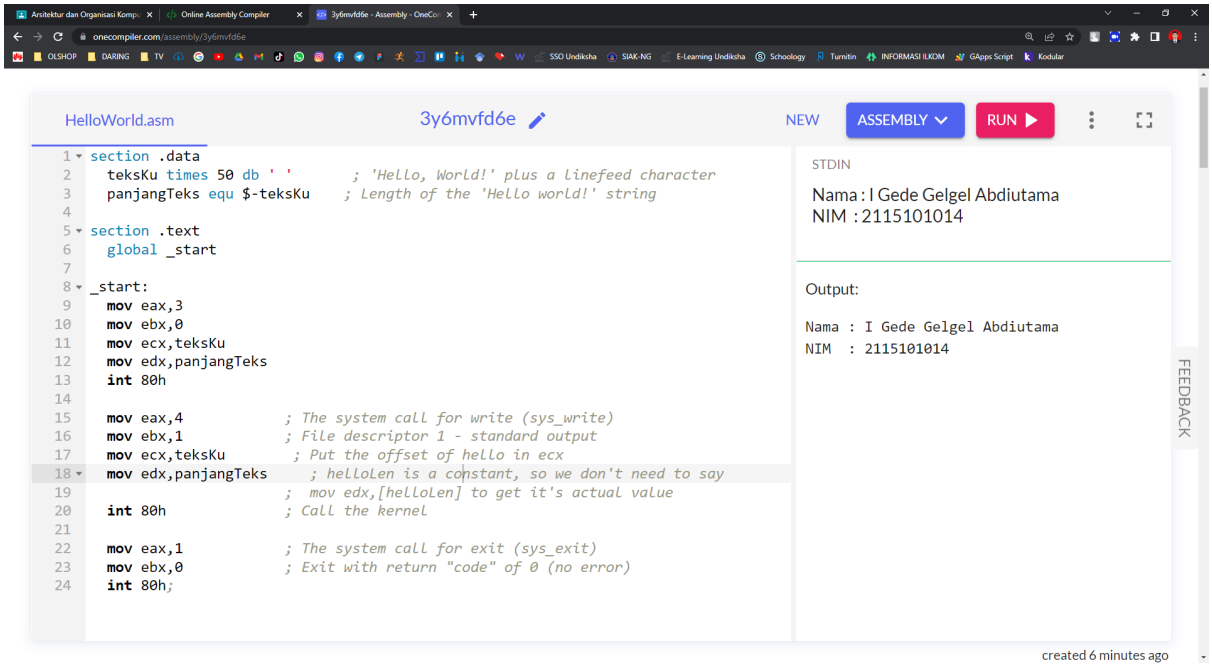
Dosen Pengampu :
I Ketut Purnamawan, S.Kom., M.Kom.

Disusun Oleh :
I Gede Gelgel Abdiutama ; 2115101014

MATA KULIAH ARSITEKTUR DAN ORGANISASI KOMPUTER
UNIVERSITAS PENDIDIKAN GANESHA
SINGARAJA
TA. 2022

A. KEGIATAN BELAJAR

Percobaan 1

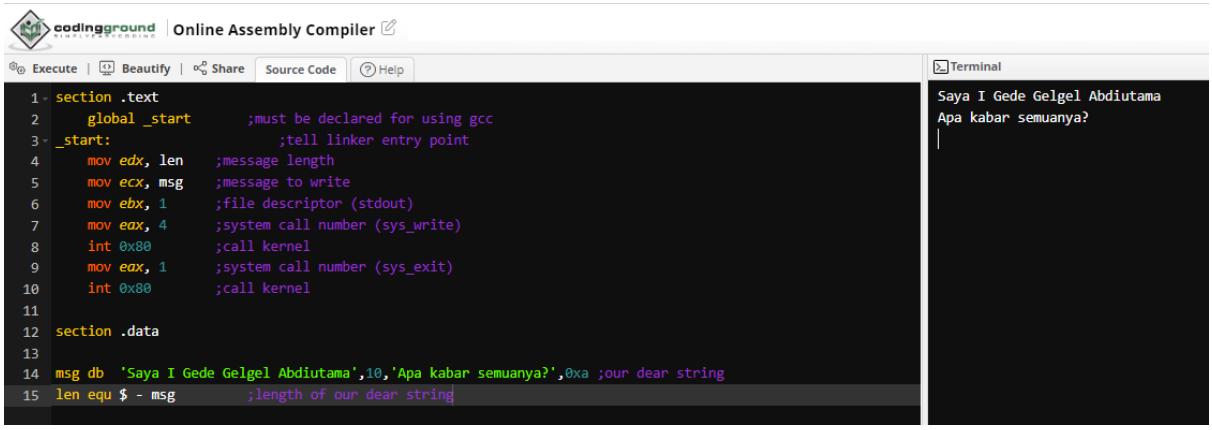



```
section .data
  teksKu times 50 db ' '       ; 'Hello, World!' plus a linefeed character
  panjangTeks equ $-teksKu     ; Length of the 'Hello world!' string

section .text
  global _start

_start:
  mov eax,3
  mov ebx,0
  mov ecx,teksKu
  mov edx,panjangTeks
  int 80h

  mov eax,4            ; The system call for write (sys_write)
  mov ebx,1            ; File descriptor 1 - standard output
  mov ecx,teksKu        ; Put the offset of hello in ecx
  mov edx,panjangTeks     ; helloLen is a constant, so we don't need to say
                       ;  mov edx,[helloLen] to get it's actual value
  int 80h              ; Call the kernel

  mov eax,1            ; The system call for exit (sys_exit)
  mov ebx,0            ; Exit with return "code" of 0 (no error)
  int 80h;
```

Kode Program :

```
section .data
  teksKu times 50 db ' '      ; 'Hello, World!' plus a linefeed character
  panjangTeks equ $-teksKu    ; Length of the 'Hello world!' string

section .text
  global _start

_start:
 mov eax,3
 mov ebx,0
 mov ecx,teksKu
 mov edx,panjangTeks
 int 80h

  mov eax,4          ; The system call for write (sys_write)
  mov ebx,1          ; File descriptor 1 - standard output
  mov ecx,teksKu       ; Put the offset of hello in ecx
  mov edx,panjangTeks    ; helloLen is a constant, so we don't need to say
                ;  mov edx,[helloLen] to get it's actual value
  int 80h           ; Call the kernel

  mov eax,1          ; The system call for exit (sys_exit)
  mov ebx,0          ; Exit with return "code" of 0 (no error)
  int 80h;
```

Percobaan 2




```
section .text
    global _start       ;must be declared for using gcc
_start:                     ;tell linker entry point
    mov edx, len    ;message length
    mov ecx, msg    ;message to write
    mov ebx, 1      ;file descriptor (stdout)
    mov eax, 4      ;system call number (sys_write)
    int 0x80        ;call kernel
    mov eax, 1      ;system call number (sys_exit)
    int 0x80        ;call kernel

section .data

msg db  'Saya I Gede Gelgel Abdiutama',10,'Apa kabar semuanya?',0xa ;our dear string
len equ $ - msg         ;length of our dear string
```

Kode Program :

```
section     .text
    global _start      ;must be declared for using gcc
_start:                ;tell linker entry point
    mov     edx, len   ;message length
    mov     ecx, msg   ;message to write
    mov     ebx, 1      ;file descriptor (stdout)
    mov     eax, 4      ;system call number (sys_write)
    int     0x80       ;call kernel
    mov     eax, 1      ;system call number (sys_exit)
    int     0x80       ;call kernel

section     .data

msg         db      'Saya I Gede Gelgel Abdiutama',10,'Apa kabar semuanya?',0xa ;our dear string
len equ     $ - msg   ;length of our dear string
```