

PRAKTIKUM MIKROPROSESOR DAN BAHASA RAKITAN

PROSEDUR DAN MAKRO

TUJUAN

1. Mahasiswa memahami dan mengerti tentang prosedur dan makro serta penggunaannya dalam bahasa mesin

DASAR TEORI

Prosedur merupakan suatu bentuk pemrograman terstruktur yang umum dijumpai dalam pemrograman bahasa level tinggi. Bahasa mesin juga memiliki suatu bentuk prosedur yang mana masih dalam bentuk yang sederhana. Meskipun demikian, prosedur dalam bahasa mesin sudah memiliki fungsi dan kegunaan yang serupa dengan prosedur pada bahasa level tinggi. Bentuk prosedur dalam bahasa mesin sebagai berikut :

NamaP	PROC	NEAR/FAR
	Isi program	
	RET	
NamaP	ENDP	

NamaP merupakan nama dari prosedurnya. Nama dari sebuah prosedurnya sebaiknya mewakili apa yang dilaksanakan oleh prosedur tersebut. PROC merupakan penanda bahwa akan dibuat sebuah prosedur. NEAR atau FAR merupakan indikasi apakah prosedur ini boleh dipanggil oleh program yang masih berada dalam satu segmen saja atau boleh berbeda segmen. Setelah prosedur dilaksanakan maka program harus kembali ke program utama yaitu dengan menggunakan RET. Untuk mengakhiri sebuah deklarasi prosedur maka dipergunakan ENDP. Untuk memanggil sebuah prosedur maka dipergunakan sintak sebagai berikut :

CALL NamaProsedur

Macro merupakan bentuk lain dari pemrograman terstruktur pada bahasa mesin. Macro memiliki bentuk sebagai berikut :

NamaM	MACRO	P1, P2, ...
	Isi macro	
	ENDM	

NamaM merupakan nama dari macro yang akan dibuat. MACRO merupakan key word yang menandakan bahwa kita akan membuat sebuah macro. P1, P2 merupakan parameter-parameter yang bisa dimasukkan ke dalam sebuah macro. Parameter ini bersifat opsional. Macro merupakan sebuah cara untuk menyingkat penulisan program dengan parameter-parameter yang bisa secara bebas dimasukkan. Jadi macro dapat dikatakan sebagai template penulisan program bahasa mesin. Oleh karena itu penggunaan macro yang berulang-ulang akan semakin memperbesar ukuran file yang akan dihasilkan.

Karena macro merupakan sebuah template penulisan bahasa mesin, maka dapat dipergunakan label sebagaimana layaknya program bahasa mesin biasa. Namun karena sifat label yang hanya boleh dipergunakan satu kali saja, maka harus dipergunakan key word LOCAL untuk membedakan satu label dengan yang lain. Seperti layaknya sebuah header file, maka macro juga dapat dibuat dalam sebuah file terpisah untuk selanjutnya diikutkan dalam program utamanya. Untuk melakukannya, maka dipergunakan sintak :

INCLUDE NamaFile

Prosedur dan macro memiliki karakteristik masing-masing sehingga harus dibedakan kapan harus menggunakan prosedur dan kapan harus menggunakan macro dalam suatu program bahasa mesin agar efisiensi programnya tetap terjaga.

PROSEDUR PERCOBAAN

A. Prosedur sederhana

1. Tuliskan program *assembly* berikut menggunakan TASM , lalu *linking* menjadi COM :

```

;~~~~~;
; PROGRAM : proc1.ASM ;
; FUNGSI : MENCETAK KARATER ;
; DENGAN PROCEDURE ;
; ;
;~~~~~;
.MODEL SMALL
.CODE
ORG 100h

Proses :
CALL Cetak_Kar ; Panggil Cetak_Kar
INT 20h

Cetak_Kar Proc Near
MOV AH,02h
MOV DL,'S'
INT 21h ; Cetak karakter
RET ; Kembali kepada si pemanggil
Cetak_Kar EndP ; END Procedures
End Proses

```

2. Jalankan programnya dan lihat hasilnya.

B. Prosedur dengan POP dan PUSH

1. Tuliskan program *assembly* berikut menggunakan TASM , lalu *linking* menjadi COM :

```
;~~~~~;
; PROGRAM : PROC2.ASM ;
; FUNGSI : MENCETAK KARATER ;
; DENGAN PROCEDURE ;
; ;
;=====;
.MODEL SMALL
.CODE
ORG 100h

TData :
    JMP Proses
    Kar db ?
    K1m db 'BATMAN SI MANUSIA KELELAWAR ' ; 28 Karakter

Proses :
    MOV CX,28 ; Banyaknya pengulangan
    XOR BX,BX ; Addressing Mode

Ulang :
    MOV DL,K1m[BX]
    MOV Kar,DL
    CALL Cetak_Kar ; Panggil Cetak_Kar
    INC BX
    LOOP Ulang
    INT 20h

Cetak_Kar Proc Near
    PUSH AX ; Simpan semua register
    PUSH DX ; Yang digunakan
    MOV AH,02h
    MOV DL,Kar
    INT 21h ; Cetak karakter
    POP DX ; Kembalikan semua register
    POP AX ; Yang disimpan
    RET ; Kembali kepada si pemanggil
Cetak_Kar EndP ; END Procedures
End TData
```

2. Jalankan programnya dan amati yang terjadi. Bandingkan jika tidak menggunakan PUSH dan POP. Apakah sama hasilnya? Mengapa bisa terjadi demikian?

C. Macro sederhana

1. Tuliskan program *assembly* berikut menggunakan TASM , lalu *linking* menjadi COM :

```
Cetak_Kar Macro Kar
    MOV CX,3
    MOV AH,02
    MOV DL,Kar
    Ulang :
    INT 21h ; Cetak Karakter
    LOOP Ulang
```

```

EndM ; End Macro

;-----;
; Program : MAC1.ASM ;
; Fungsi : Menggunakan Macro ;
; Untuk mencetak ;
; huruf 'SSS' ;
;-----;
    .Model Small
    .Code
    ORG 100h
Proses:
    Cetak_Kar 'S' ; Cetak Huruf S
    INT 20h
End Proses

```

2. Jalankan programnya dan amati yang terjadi.

D. Macro dengan label

1. Tuliskan program *assembly* berikut menggunakan TASM , lalu *lingking* menjadi COM :

```

Cetak_Kar Macro Kar
    Local Ulang ; Label 'Ulang' jadikan Local
    MOV CX,3
    MOV AH,02
    MOV DL,Kar
    Ulang:
    INT 21h ; Cetak Karakter
    LOOP Ulang
EndM ; End Macro

```

```

;-----;
; Program : MAC2.ASM ;
; Fungsi : Menggunakan Macro ;
; Untuk mencetak ;
; huruf 'PPPCC' ;
;-----;
    .Model Small
    .Code
    ORG 100h
Proses:
    Cetak_Kar 'P' ; Cetak Huruf P
    Cetak_Kar 'C' ; Cetak Huruf C
    INT 20h
End Proses

```

2. Jalankan programnya dan amati yang terjadi. Ubah label Proses menjadi Ulang dan amati yang terjadi!

E. Pustaka macro

1. Tuliskan program berikut dan simpan sebagai macku.mcr di dalam folder yang sama dengan TASM.EXE :

```

;/=====\  

; Program : macku.MCR ;  

;\=====/  

Cetak_Kar MACRO Kar ; Macro untuk mencetak

```

```

MOV AH,02 ; Karakter
MOV DL,Kar
INT 21h
ENDM

```

```

Cetak_Klm MACRO Klm ; Macro untuk mencetak
LEA DX,Klm ; kalimat
MOV AH,09
INT 21h
ENDM

```

2. Tuliskan program *assembly* berikut menggunakan TASM , lalu *linking* menjadi COM :

```

; /===== \;
; Program : mac3.ASM ;
; Fungsi : Contoh menggunakan ;
; pustaka macro ;
; \===== \;
INCLUDE macku.mcr ; Gunakan file macku.mcr
.MODEL small
.CODE
ORG 100h
TData :
JMP Proses
Ka10 db 'PENGUNAAN PUSTAKA Macro $'
Proses:
Cetak_Klm Ka10 ; Cetak Kalimat Ka10
Cetak_Kar 'Y' ; Cetak Huruf 'Y'
INT 20h
End TData

```

3. Jalankan programnya dan amati yang terjadi.

TUGAS

1. Sebutkan 5 perbedaan prosedur dan macro?