

Mesin Abstrak dan Mesin Karakter

IF2110/IF2111 – Algoritma dan Struktur Data
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Mesin

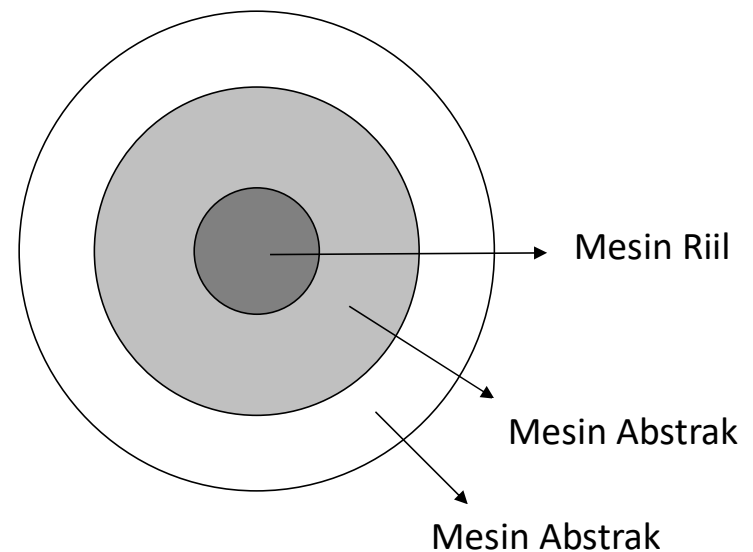
Mesin:

mekanisme yang terdefinisi dan mengerti serta mampu untuk **mengeksekusi aksi-aksi primitif** yang terdefinisi untuk mesin tersebut

Mesin abstrak:

mesin yang **dianggap ada** dan diasumsikan mampu melakukan mekanisme yang didefinisikan untuk mesin tersebut

Mesin abstrak memodelkan suatu semesta (*universe*) tertentu



Mesin Abstrak

Mesin abstrak mendefinisikan:

- Sekumpulan **state** yang mungkin
- Sekumpulan **aksi primitif** yang diasumsikan dapat dimengerti dan dieksekusi mesin yang bersangkutan

Contoh mesin abstrak:

mesin gambar

mesin integer

mesin rekam

mesin karakter

Mesin Karakter

Mesin Karakter (1)

Terdiri atas:

- Pita berisi deret karakter, diakhiri dengan **MARK** berupa '.' (titik)

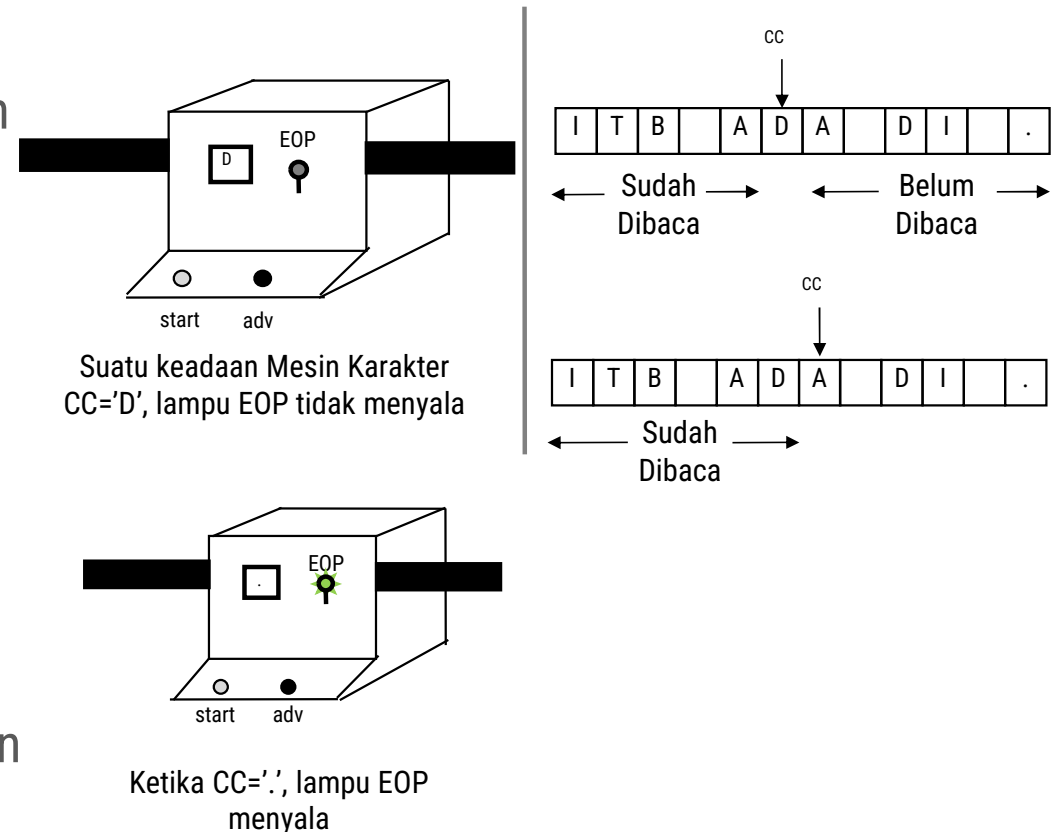
Pita hanya berisi MARK → **pita kosong**

- Tombol **start** dan **adv**
- "**Jendela**" → ukuran sebesar satu karakter

CC (*Current Character*) → karakter yang sedang tampak di jendela

- Lampu **EOP** (*End of Pita*)

State mesin karakter ditentukan oleh **CC** dan **EOP**



Mesin Karakter (2)

Primitif terkait posisi pita

procedure start

```
{ Mesin siap dioperasikan. Pita disiapkan untuk dibaca.  
  Karakter pertama yang ada pada pita posisinya adalah pada  
  jendela  
  I.S.: sembarang  
  F.S.: cc adalah karakter pertama pada pita  
        Jika cc ≠ MARK maka eop akan padam (false)  
        Jika cc = MARK maka eop akan menyala (true) }
```

procedure adv

```
{ Pita dimajukan satu karakter.  
  I.S.: Karakter pada jendela = cc, cc ≠ MARK  
  F.S.: cc adalah karakter berikutnya dari cc yang lama,  
        cc mungkin = MARK  
        Jika cc = MARK maka eop akan menyala (true) }
```

EOP diwakili oleh boolean, bernilai **true** jika menyala; atau **false** jika tidak menyala. Jika **EOP** menyala, mesin sudah tidak dapat dioperasikan lagi.

Studi Kasus Mesin Karakter (1) - CountKarakter

Diberikan sebuah mesin karakter dengan pita berisi karakter (mungkin kosong). Buatlah algoritma untuk menghitung banyaknya huruf yang ada pada pita tersebut. Banyaknya karakter pada pita kosong adalah nol.

Program CountCharacters

{ SKEMA PEMROSESAN DENGAN MARK:

menghitung banyaknya huruf pada pita karakter }

KAMUS

```
ctr: integer
```

ALGORITMA

```
ctr ← 0           { Inisialisasi }
```

```
start      { First Elmt }
```

```
while (cc  $\neq$  MARK) do { not EOP }
```

```
ctr ← ctr + 1    { Proses }
```

```
adv      { Next Elmt }
```

$\{ CC = MARK \}$

```
output(ctr)           { Terminasi }
```

Studi Kasus Mesin Karakter (2) - Hitung-A

Diberikan sebuah mesin karakter dengan pita berisi karakter (mungkin kosong). Buatlah algoritma untuk menghitung banyaknya huruf 'A' yang ada pada pita tersebut. Banyaknya karakter 'A' pada pita kosong adalah nol.

Program CountA

{ *SKEMA PEMROSESAN DENGAN MARK:*
menghitung banyaknya huruf A pada pita karakter }

KAMUS

ctr: integer

ALGORITMA

ctr ← 0	{ <i>Inisialisasi, CI = 0</i> }
start	{ <i>First Elmt</i> }
<u>while</u> (cc ≠ MARK) <u>do</u>	{ <i>not EOP</i> }
<u>if</u> cc = 'A' <u>then</u>	{ <i>Proses</i> }
ctr ← ctr + 1	
adv	{ <i>Next Elmt</i> }
{ cc = MARK }	
<u>output</u> (ctr)	{ <i>Terminasi</i> }

Mesin Karakter

Dalam Bahasa C

mesinkar.h

```
#ifndef __MESIN_KAR__
#define __MESIN_KAR__

#include "boolean.h"

#define MARK '.'

/* State Mesin */
extern char cc;
extern boolean eop;

void start();
/* Mesin siap dioperasikan. Pita disiapkan untuk dibaca.
   Karakter pertama yang ada pada pita posisinya adalah pada jendela.
   I.S.: sembarang
   F.S.: cc adalah karakter pertama pada pita
         Jika cc != MARK maka eop akan padam (false)
         Jika cc = MARK maka eop akan menyala (true) */
void adv();
/* Pita dimajukan satu karakter.
   I.S.: Karakter pada jendela = cc, cc != MARK
   F.S.: cc adalah karakter berikutnya dari cc yang lama,
         cc mungkin = MARK
         Jika cc = MARK maka eop akan menyala (true) */

#endif
```

mesinkar.c

```
#include <stdio.h>
#include "mesinkar.h"

char cc;
boolean eop;

static FILE *pita;
static int retval;

void start() {
    /* Mesin siap dioperasikan. Pita disiapkan ... */
    /* Algoritma */
    pita = fopen("pitakar.txt", "r");
    adv();
}

void adv() {
    /* Pita dimajukan satu karakter. ... */
    /* Algoritma */
    retval = fscanf(pita, "%c", &cc);
    eop = (cc == MARK);
    if (eop) {
        fclose(pita);
    }
}
```

Latihan-1: Hitung-LE

Diberikan sebuah mesin karakter dengan pita berisi karakter (mungkin kosong), Buatlah algoritma untuk menghitung banyaknya pasangan huruf 'L' dan 'E' yang ada pada pita tersebut. Banyaknya pasangan huruf 'L' dan 'E' pada pita kosong adalah nol.

function HITUNGLE (input s: string) → integer

KAMUS LOKAL

count : integer

ALGORITMA

count ← 0

start

while (cc != MARK) do

if (cc = 'L') then

adv

if (cc = 'E') then

count ← count + 1

else

adv

else

adv