



# Mesin Karakter dan Mesin Kata (Bag. 1)

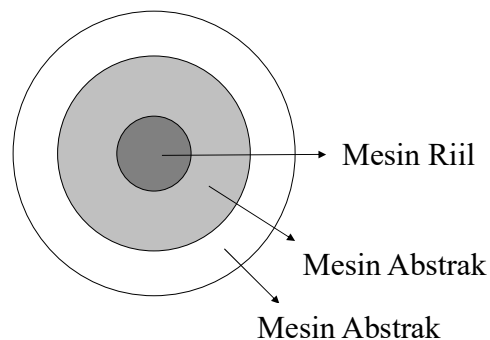
Tim Pengajar  
IF2110/Algoritma dan Struktur Data  
Sem. I 2017/2018

IF2110/sem. 1 2017/2018



## 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



IF2110/sem. 1 2017/2018



# Mesin Abstrak

- Mendefinisikan mesin abstrak berarti 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**

IF2110/sem. 1 2017/2018



# Mesin Karakter

•IF2110/sem. 1 2017/2018

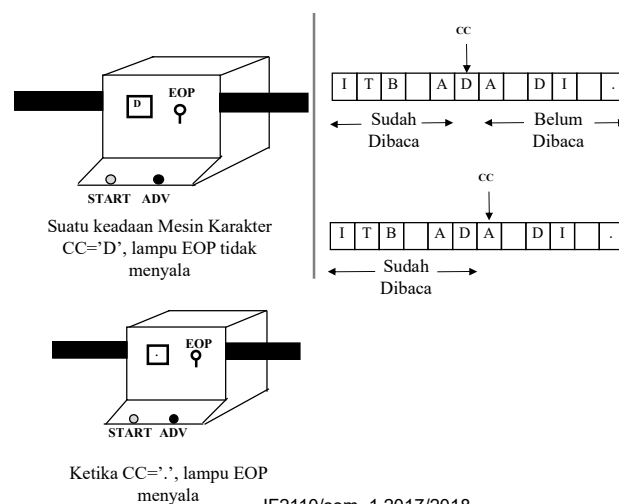
# Mesin Karakter (1)



- Mesin karakter adalah mesin abstrak yang terdiri atas:
  - Pita berisi deret karakter, diakhiri dengan '.' (titik)
    - Pita yang hanya berisi '.' disebut sebagai pita kosong,
  - Tombol **START** dan **ADV**
  - Lampu **EOP** (End Of Pita)
  - "Jendela" yang ukurannya sebesar satu karakter
    - Hanya karakter yang posisinya sedang pada jendela dapat dibaca; karakter lain tidak kelihatan
    - Karakter yang sedang pada jendela dinamakan **CC** (Current Character)
- State mesin karakter ditentukan oleh CC dan EOP

IF2110/sem. 1 2017/2018

# Mesin Karakter (2)



IF2110/sem. 1 2017/2018

## Mesin Karakter (3)



- 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 ≠ '.' maka EOP akan padam (false)  
         Jika CC = '.' maka EOP akan menyala (true) }
```

### procedure ADV

```
{ Pita dimajukan satu karakter.  
  I.S. : Karakter pada jendela = CC, CC ≠ '.'  
  F.S. : CC adalah karakter berikutnya dari CC yang lama,  
         CC mungkin = '.'  
         Jika CC = '.' 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.

IF2110/sem. 1 2017/2018

## Studi Kasus Mesin Karakter (1) CountHuruf



- 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 COUNTHURUF

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

### KAMUS

CI : integer

### ALGORITMA

```
CI ← 0           { Inisialisasi }  
START           { First Elmt }  
while (CC ≠ '.') do { not EOP }  
    CI ← CI + 1   { Proses }  
    ADV           { Next_Elmt }  
{ CC = '.' }  
output (CI)     { Terminasi }
```

IF2110/sem. 1 2017/2018

## 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 COUNT_A
{ SKEMA PEMROSESAN DENGAN MARK :
menghitung banyaknya huruf A pada pita karakter }
KAMUS
  CI : integer
ALGORITMA
  CI ← 0          { Inisialisasi, CI = 0 }
  START          { First_Elmt }
  while (CC ≠ '.') do { not EOP }
    depend on CC    { Proses }
    CC = 'A' : CI ← CI + 1
    CC ≠ 'A' : -
  ADV           { Next_Elmt }
  { CC = '.' }
  output (CI)     { Terminasi }
```

IF2110/sem. 1 2017/2018

## Mesin Karakter

Dalam Bahasa C



IF2110/sem. 1 2017/2018

# 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
```

•IF2110/sem. 1 2017/2018

# 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 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)
   */

/* Algoritma */
pita = fopen("pitakar.txt", "r");
ADV();
}

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) */

/* Algoritma */
retval = fscanf(pita, "%c", &CC);
EOP = (CC == MARK);
if (EOP) {
    fclose(pita);
}
}
```

•IF2110/sem. 1 2017/2018

# 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.

•IF2110/sem. 1 2017/2018

# Mesin Kata



•IF2110/sem. 1 2017/2018

# Mesin Kata (1)



- Mesin Kata:  
Mesin abstrak yang bekerja memproses kata berdasarkan mesin karakter
- Diberikan sebuah mesin karakter dengan pita berisi karakter (mungkin kosong), yang diakhiri titik ('.')

IF2110/sem. 1 2017/2018

# Mesin Kata (2)



- Kata:
  - sederetan karakter suksesif pada pita yang merupakan karakter bukan *blank*
- Definisi type Kata:

```
type Kata : <TabKata : array [1..NMax] of character,  
             Length : integer >  
{ TabKata adalah tempat penampung/container kata,  
  Length menyatakan panjangnya kata }  
{ Deklarasi: K : Kata }  
{ Definisi kata kosong: K.Length = 0 }  
{ Memori array yg dipakai selalu dimulai dari indeks ke-1 }
```

IF2110/sem. 1 2017/2018





## Mesin Kata (3)

- Model-model akuisisi KATA (token) pada pita karakter: versi 1, 2, dan 3

a. Hanya mengandung titik (pita kosong)

•

b. Hanya mengandung blank diakhiri titik

—•

c. Mengandung blank di awal dan akhir pita

—•

d. Tidak mengandung blank di awal maupun di akhir pita

—•

e. Mengandung blank di akhir pita

—•

f. Mengandung blank di awal pita

—•

IF2110/sem. 1 2017/2018



## Mesin Kata

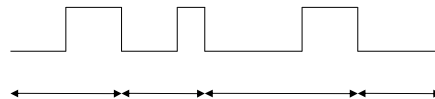
### Model Akuisisi Kata Versi 1

•IF2110/sem. 1 2017/2018

# Model Akuisisi Kata Versi 1



- Kata diakuisisi mulai dari karakter pertama sesudah akhir kata (atau karakter pertama pita untuk kata pertama)



- Akhir dari proses adalah sebuah boolean (**EndKata**), yang akan berisi true jika kata terakhir telah diakuisisi dan diproses
- Akuisisi kata terakhir menghasilkan 'kata kosong'.

IF2110/sem. 1 2017/2018

## Model Akuisisi Kata - Versi 1 (1)



### KAMUS UMUM

```
{ ***** Mesin lain yang dipakai ***** }
use MESINKAR

{ ***** Konstanta ***** }
constant MARK : character = '.'
constant BLANK : character = ' '
constant NMax : integer = 50 { jumlah maksimum karakter suatu kata }

{ ***** Definisi Type Kata ***** }
type Kata : < TabKata : array [1..NMax] of character,
              Length : integer >
{ TabKata adalah tempat penampung/container kata,
  Length menyatakan panjangnya kata }

{ ***** Definisi State Mesin Kata ***** }
EndKata : boolean { penanda akhir akuisisi kata }
CKata : Kata { kata yang sudah diakuisisi dan akan diproses }
```

IF2110/sem. 1 2017/2018

## Model Akuisisi Kata - Versi 1 (2)



```
{***** Primitif-Primitif Mesin Kata *****}
procedure Ignore_Blank
{ Mengabaikan satu atau beberapa BLANK }
{ I.S. : CC sembarang }
{ F.S. : CC ≠ BLANK atau CC = MARK }
procedure STARTKATA
{ I.S. : CC sembarang }
{ F.S. : EndKata = true, dan CC = Mark; }
{
    atau EndKata = false, CKata adalah kata yang sudah
    diakuisisi,
    CC karakter pertama sesudah karakter terakhir kata }
procedure ADVKATA
{ I.S. : CC adalah karakter pertama kata yang akan diakuisisi }
{ F.S. : CKata adalah kata terakhir yang sudah diakuisisi,
    CC adalah karakter pertama sesudah karakter terakhir kata }
{ Proses : Akuisisi kata menggunakan procedure SalinKata }
procedure SalinKata
{ Mengakuisisi kata, menyimpan dalam CKata }
{ I.S. : CC adalah karakter pertama dari kata }
{ F.S. : CKata berisi kata yang sudah diakuisisi; CC = BLANK atau CC =
MARK; CC adalah karakter sesudah karakter terakhir yang diakuisisi }
```

IF2110/sem. 1 2017/2018

## Model Akuisisi Kata - Versi 1 (3)



```
procedure Ignore_Blank
{ Mengabaikan satu atau beberapa BLANK }
{ I.S. : CC sembarang }
{ F.S. : CC ≠ BLANK atau CC = MARK }
Kamus Lokal

Algoritma
    while (CC = BLANK) and (CC ≠ MARK) do
        ADV
    { CC ≠ BLANK or CC = MARK }
```

IF2110/sem. 1 2017/2018

## Model Akuisisi Kata - Versi 1 (4)



### **procedure** STARTKATA

```
{ I.S. : CC sembarang }  
{ F.S. : EndKata = true, dan CC = Mark; }  
{  
    atau EndKata = false, CKata adalah kata yang sudah  
    diakuisisi,  
    CC karakter pertama sesudah karakter terakhir kata }
```

### **Kamus Lokal**

### **Algoritma**

```
START  
Ignore_Blank  
depend on (CC)  
    CC = MARK : EndKata ← true  
    CC ≠ MARK : EndKata ← false  
    SalinKata
```

IF2110/sem. 1 2017/2018

## Model Akuisisi Kata - Versi 1 (5)



### **procedure** ADVKATA

```
{ I.S. : CC adalah karakter pertama kata yang akan diakuisisi }  
{ F.S. : CKata adalah kata terakhir yang sudah diakuisisi,  
    CC adalah karakter pertama dari kata berikutnya,  
    mungkin MARK }  
{ Proses : Akuisisi kata menggunakan procedure SalinKata }
```

### **Kamus Lokal**

### **Algoritma**

```
Ignore_Blank  
depend on (CC)  
    CC = MARK : EndKata ← true  
    CC ≠ MARK : SalinKata
```

IF2110/sem. 1 2017/2018

## Model Akuisisi Kata - Versi 1 (6)



### **procedure** SalinKata

```
{ Mengakuisisi kata, menyimpan dalam CKata }  
{ I.S. : CC adalah karakter pertama dari kata }  
{ F.S. : CKata berisi kata yang sudah diakuisisi; CC = BLANK atau  
CC = MARK; CC adalah karakter sesudah karakter terakhir yang  
diakuisisi }
```

### **Kamus Lokal**

i : integer

### **Algoritma**

```
i ← 1  
iterate  
  CKata.TabKata[i] ← CC  
  ADV  
stop : (CC = MARK) or (CC = BLANK)  
  i ← i + 1  
{ CC = MARK or CC = BLANK }  
CKata.Length ← i
```

IF2110/sem. 1 2017/2018

## Studi Kasus 1 Panjang Rata-Rata Kata



- Diberikan pita berisi karakter (mungkin kosong), yang diakhiri titik, hitunglah panjang rata-rata kata yang ada pada pita tsb. Panjang kata rata-rata tidak terdefinisi jika pita kosong atau pita tidak mengandung kata (hanya berisi '*blank*' dan titik).

IF2110/sem. 1 2017/2018

# Panjang Rata-Rata Kata Model Akuisisi Kata Versi 1 (1)



```
Program PanjangRataRataKata1
{ Menghitung panjang rata-rata kata dalam pita karakter }
{ Model akuisisi kata versi 1 }

KAMUS
{ *** Mesin yang digunakan *** }
    USE MesinKata1

    NbKata : integer    { banyaknya kata dalam pita }
    LTotal : integer    { akumulasi panjang kata }

ALGORITMA
{ di balik }
```

•IF2110/sem. 1 2017/2018

# Panjang Rata-Rata Kata Model Akuisisi Kata Versi 1 (2)



```
ALGORITMA
LTotal ← 0
NbKata ← 0
STARTKATA
while not EndKata do
    LTotal ← LTotal + CKata.Length
    NbKata ← NbKata + 1
    ADVKATA
{ EndKata = true: semua karakter sudah diakuisisi }
if (NbKata ≠ 0) then
    output (LTotal/NbKata)
else { NbKata = 0 }
    output ("Pita tidak mengandung kata")
```

•IF2110/sem. 1 2017/2018

## Studi Kasus 2 Hitung WHILE



- Diberikan suatu pita karakter yang mengandung abjad, blank, dan diakhiri titik, harus dicari banyaknya kemunculan kata 'WHILE' pada pita tersebut

IF2110/sem. 1 2017/2018

## Hitung WHILE Model Akuisisi Kata Versi 1 (1)



```
Program HitungWhile1
{ Menghitung banyaknya kata WHILE dalam pita karakter }
{ Model akuisisi kata versi 1 }

KAMUS
{ *** Mesin yang digunakan *** }
    USE MesinKata1

    KataWHILE : Kata    { Kata yang menyimpan WHILE }
    NWHILE : integer    { banyaknya kata WHILE }

    function IsKataSama (K1, K2 : Kata) → boolean
    { Menghasilkan true jika K1 = K2 }

ALGORITMA
{ di balik }
```

•IF2110/sem. 1 2017/2018

# Hitung WHILE

## Model Akuisisi Kata Versi 1 (2)



### ALGORITMA

```
{ Inisialisasi KataWHILE }
KataWHILE.TabKata[1] ← 'W'
KataWHILE.TabKata[2] ← 'H'
KataWHILE.TabKata[3] ← 'I'
KataWHILE.TabKata[4] ← 'L'
KataWHILE.TabKata[5] ← 'E'
KataWHILE.Length ← 5
NWHILE ← 0
STARTKATA
while not EndKata do
  if IsKataSama(KataWHILE, CKata) then
    NWHILE ← NWHILE + 1
  ADVKATA
{ EndKata = true: semua karakter sudah diakuisisi }
output (NWHILE)
```

•IF2110/sem. 1 2017/2018

# Hitung WHILE

## Model Akuisisi Kata Versi 1 (3)



**function** IsKataSama (K1, K2 : Kata) → boolean  
{ Menghasilkan true jika K1 = K2 }

### Kamus Lokal

i : integer

### Algoritma

```
if K1.Length ≠ K2.Length then
  → false
else { K1.Length = K2.Length }
  i ← 1
  while (i < K1.Length) and (K1.TabKata[i] = K2.TabKata[i]) do
    i ← i + 1
  { i ≥ K1.Length or K1.TabKata[i] ≠ K2.TabKata[i] }
  → K1.TabKata[i] = K2.TabKata[i]
```

•IF2110/sem. 1 2017/2018





# Mesin Kata

## Model Akuisisi versi 2 dan 3

- Model akuisisi versi 2: seperti versi 1, namun akhir dari proses akuisisi adalah kata 'kosong'
- Model akuisisi versi 3: mengabaikan blank pada awal pita dan memproses sisanya → model akuisisi tanpa mark

•IF2110/sem. 1 2017/2018



# Mesin Kata

Dalam Bahasa C

IF2110/sem. 1 2017/2018

## mesinkata1.h (model akuisisi v.1)



```
#ifndef __MESINKATA1_H__
#define __MESINKATA1_H__

#include "mesinkar.h"

#define NMax 50
#define BLANK ' '

typedef struct {
    char TabKata[NMax+1];
    int Length;
} Kata;

/* State Mesin Kata */
extern boolean EndKata;
extern Kata CKata;

void IgnoreBlank();
/* Mengabaikan satu atau beberapa BLANK
   I.S. : CC sembarang
   F.S. : CC ≠ BLANK atau CC = MARK */
```

•IF2110/sem. 1 2017/2018

## mesinkata1.h (model akuisisi v.1)



```
void STARTKATA();
/* I.S. : CC sembarang
   F.S. : EndKata = true, dan CC = Mark;
          atau EndKata = false, CKata adalah kata yang sudah
          diakuisisi,
          CC karakter pertama sesudah karakter terakhir kata */

void ADVKATA();
/* I.S. : CC adalah karakter pertama kata yang akan diakuisisi
   F.S. : CKata adalah kata terakhir yang sudah diakuisisi,
          CC adalah karakter pertama dari kata berikutnya,
          mungkin MARK
   Proses : Akuisisi kata menggunakan procedure SalinKata */

void SalinKata();
/* Mengakuisisi kata, menyimpan dalam CKata
   I.S. : CC adalah karakter pertama dari kata
   F.S. : CKata berisi kata yang sudah diakuisisi;
          CC = BLANK atau CC = MARK;
          CC adalah karakter sesudah karakter terakhir yang
          diakuisisi */
#endif
```

•IF2110/sem. 1 2017/2018

## mesinkata1.c (model akuisisi v.1)



```
#include "mesinkata1.h"

boolean EndKata;
Kata CKata;

void IgnoreBlank()
/* Mengabaikan satu atau beberapa BLANK
   I.S. : CC sembarang
   F.S. : CC ≠ BLANK atau CC = MARK */
{ /* Kamus Lokal */

    /* Algoritma */
    while ((CC == BLANK) && (CC != MARK)) {
        ADV();
    } /* CC != BLANK or CC = MARK */
}
```

•IF2110/sem. 1 2017/2018

## mesinkata1.c (model akuisisi v.1)



```
void STARTKATA()
/* I.S. : CC sembarang
   F.S. : EndKata = true, dan CC = Mark;
          atau EndKata = false, CKata adalah kata yang sudah
          diakuisisi,
          CC karakter pertama sesudah karakter terakhir kata */
{ /* Kamus Lokal */

    /* Algoritma */
    START();
    IgnoreBlank();
    if (CC == MARK) {
        EndKata = true;
    } else /* CC != MARK */ {
        EndKata = false;
        SalinKata();
    }
}
```

•IF2110/sem. 1 2017/2018

## mesinkata1.c (model akuisisi v.1)



```
void ADVKATA()
/* I.S. : CC adalah karakter pertama kata yang akan diakuisisi
   F.S. : CKata adalah kata terakhir yang sudah diakuisisi,
          CC adalah karakter pertama dari kata berikutnya,
          mungkin MARK
   Proses : Akuisisi kata menggunakan procedure SalinKata */
{ /* Kamus Lokal */

    /* Algoritma */
    IgnoreBlank();
    if (CC == MARK) {
        EndKata = true;
    } else /* CC != MARK */ {
        SalinKata();
    }
}
```

•IF2110/sem. 1 2017/2018

## mesinkata1.c (model akuisisi v.1)



```
void SalinKata()
/* Mengakuisisi kata, menyimpan dalam CKata
   I.S. : CC adalah karakter pertama dari kata
   F.S. : CKata berisi kata yang sudah diakuisisi;
          CC = BLANK atau CC = MARK;
          CC adalah karakter sesudah karakter terakhir yang
          diakuisisi */
{ /* Kamus Lokal */
    int i = 1; /* inisialisasi */
    /* Algoritma */
    for (;;) {
        CKata.TabKata[i] = CC;
        ADV();
        if ((CC == MARK) || (CC == BLANK)) {
            break;
        } else {
            i++;
        }
    } /* CC = MARK or CC = BLANK */
    CKata.Length = i;
}
```

•IF2110/sem. 1 2017/2018

# mainkata.c



```
#include "mesinkatal.h"

int main() {
    STARTKATA();
    while (!EndKata) {
        for (i=1;i<=CKata.Length;i++) {
            printf("%c",CKata.TabKata[i]);
        }
        printf("\n");
        ADVKATA();
    }
    return 0;
}
```

•IF2110/sem. 1 2017/2018

## Cara Kompilasi



```
$ cc -c mesinkar.c
$ cc -c mesinkata.c
$ cc -c mainkata.c
$ cc -o mainkata mesinkar.o mesinkata.o mainkata.o
```

Atau cara lain:

```
$ cc -o mainkata mesinkar.c mesinkata.c mainkata.c
```

•IF2110/sem. 1 2017/2018

# PR



- Modul pra-praktikum:
  - MESIN KATA: Bagian 1. Model Akuisisi Versi-1
  - Perhatikan bahwa procedure SalinKata yang digunakan mempertimbangkan bahwa jika Kata melebihi Nmax, maka Kata akan dipotong
- Header file mesinkata.h akan disediakan mulai Selasa, 19 September 2017
- Dikumpulkan pada link tugas pra-praktikum yang akan disediakan hingga Kamis, 28 September 2017 jam 11.00
- Untuk digunakan dalam praktikum Kamis, 28 Sept 2017

IF2110/sem. 1 2017/2018

# Kuis-1



- Dilaksanakan pada jam kuliah pada hari Selasa, 19 September 2017
- Materi: ADT biasa, ADT Array
- Notasi Algoritmik
- Bawa: pensil + penghapus, bolpoin (hitam/biru)