



---

# ADT dalam Bahasa C

IF2110 – Semester I 2020/2021

---

Tim Pengajar IF2110

# Tujuan

---

- Mahasiswa dapat mengimplementasikan ADT dengan memakai modul program dalam bahasa C

# Contoh ADT POINT – Notasi algoritmik

```
{*** Definisi Abstract Data Type point ***}  
type point: < x: real, { absis }  
          y: real { ordinat } >
```

```
{*** Konstruktor POINT ***}  
function MakePoint (x: real, y: real) → point  
{ membentuk sebuah point dari x dan y dengan x sebagai absis dan }  
{ y sebagai ordinat }
```

```
{*** Selektor POINT ***}  
{*** Get komponen point ***}  
function GetAbsis (P: point) → real  
{ mengirimkan komponen absis dari P }  
function GetOrdinat (P: point) → real  
{ mengirimkan komponen ordinat dari P }  
{*** Set nilai komponen point ***}  
procedure SetAbsis (input/output P: point, input newX: real)  
{ Mengubah nilai komponen Absis dari P }  
procedure SetOrdinat (input/output P: point, input newY: real)  
{ Mengubah nilai komponen Ordinat dari P }
```

# Contoh ADT POINT - Notasi algoritmik (lanj.)

{\*\*\* Prosedur - Interaksi dengan I/O device, BACA/TULIS \*\*\*}

procedure BacaPOINT (output P: point)

{ Membentuk P dari x dan y yang dibaca }

procedure TulisPOINT (input P: point)

{ Nilai P ditulis ke layar dgn format "(X,Y)" }

{\*\*\* Fungsi – Operasi lain terhadap type data \*\*\*}

function geser (P1: point, dx, dy: real) → point

{ Menghasilkan point yang merupakan hasil penggeseran P sebesar }

{ dx arah sumbu x dan dy arah sumbu y }

{\*\*\* Predikat \*\*\*}

function IsOrigin? (P: point) → boolean

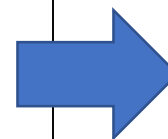
{ Mengirimkan nilai benar jika P adalah titik origin yaitu }

{ titik <0,0> }

## Catatan Khusus: Selektor ADT dalam Bahasa C

- Selektor Get dan Set pada Bahasa C dapat diganti dengan C preprocessor/macro berparameter

```
{*** Selektor POINT ***}  
  
{*** Get komponen point ***}  
function GetAbsis (P: point) → real  
{ mengirimkan komponen absis dari P }  
function GetOrdinat (P: point) → real  
{ mengirimkan komponen ordinat dari P }  
  
{*** Set nilai komponen point ***}  
procedure SetAbsis (input/output P: point, input newX: real)  
{ Mengubah nilai komponen Absis dari P }  
procedure SetOrdinat (input/output P: point, input newY: real)  
{ Mengubah nilai komponen Ordinat dari P }
```



```
#define Absis(P) (P).x  
#define Ordinat(P) (P).y
```

# ADT dalam Bahasa C, file point.h

```
#ifndef POINT_H  
#define POINT_H
```

```
#include "boolean.h"
```

Include **boolean.h**  
sehingga type boolean  
dikenal

```
/**** Definisi Abstract Data Type POINT ****/  
typedef struct {  
    float x; /* absis */  
    float y; /* ordinat */  
} point;
```

```
/**** Selektor point ****/  
#define Absis(P) (P).x  
#define Ordinat(P) (P).y
```

Notasi akses:  
Menggantikan  
fungsi/prosedur selektor  
get dan set

```
/**** Konstruktor POINT ****/  
point MakePoint (float x, float y);  
/* Membentuk sebuah point dari x dan y dengan x sebagai absis dan y sebagai ordinat */  
...
```

# ADT Point dalam Bahasa C, file point.h (lanj.)

---

```
...
**** Predikat ****/
boolean IsOrigin (point P);
/* Mengirimkan nilai benar jika P adalah titik origin yaitu titik <0,0> */

**** Prosedur - Interaksi dengan I/O device, BACA/TULIS ****/
void BacaPOINT (point * P);
/* Membentuk P dari x dan y yang dibaca dari keyboard */
void TulisPOINT (point P);
/* Nilai P ditulis ke layar dg format "(X,Y)" */

**** Fungsi/Operasi lain terhadap point ****/
point geser (point P, float dx, float dy);
/* Menghasilkan point yang merupakan hasil penggeseran P */
/* sebesar dx arah sumbu x dan dy arah sumbu y */

#endif
```

# Type Boolean dalam Bahasa C

## Isi File: boolean.h

---

```
/* File: boolean.h */  
/* Definisi type boolean */  
  
#ifndef _BOOLEAN_h  
#define _BOOLEAN_h  
  
#define boolean unsigned char  
#define true 1  
#define false 0  
  
#endif
```



# ADT Point dalam Bahasa C

## File point.c

---

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

Include file header

```
/**** Konstruktor POINT ****/
point MakePoint (float x, float y)
/* Membentuk sebuah point dari x dan y dengan x sebagai absis dan y sebagai ordinat */
{ /* KAMUS */
    point P;
    /* ALGORITMA */
    Absis(P) = x;
    Ordinat(P) = y;
    return P;
}
...
```

Contoh Penggunaan  
Selektor

# ADT Point dalam Bahasa C

## File point.c (lanj.)

...

```
/**** Predikat ****/  
boolean IsOrigin (point P)  
/* Mengirimkan nilai benar jika P adalah titik origin yaitu titik <0,0> */  
{ /* ALGORITMA */  
    return ((Absis(P) == 0) && (Ordinat(P) == 0));  
}
```

Contoh Penggunaan  
Selektor

...

Hindari:

```
if ((Absis(P) == 0) && (Ordinat(P) == 0)) {  
    return true;  
} else {  
    return false;  
}
```

# ADT Point dalam Bahasa C

## File point.c (lanj.)

---

...

```
/**** Prosedur - Interaksi dengan I/O device, BACA/TULIS ****/  
void BacaPOINT (point * P)  
/* Membentuk P dari x dan y yang dibaca dari keyboard */  
{ /* KAMUS */  
    float x, y;  
    /* ALGORITMA */  
    scanf("%f", &x);  
    scanf("%f", &y);  
    *P = MakePoint(x,y);  
}
```

...

Contoh Penggunaan  
Konstruktor

# ADT Point dalam Bahasa C

## File point.c (lanj.)

---

...

```
void TulisPOINT (point P)
```

```
/* Nilai P ditulis ke layar dg format "(X,Y)" */
```

```
{ /* ALGORITMA */
```

```
    printf("(%.2f,%.2f)", Absis(P), Ordinat(P));
```

```
}
```

```
/**** Fungsi/Operasi lain terhadap point ****/
```

```
point geser (point P, float dx, float dy)
```

```
/* Menghasilkan point yang merupakan hasil penggeseran P */
```

```
/* sebesar dx arah sumbu x dan dy arah sumbu y */
```

```
{ /* ALGORITMA */
```

```
    return (MakePoint(Absis(P)+dx, Ordinat(P)+dy));
```

```
}
```

# Driver ADT Point

## File: mpoint.c

---

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

```
int main () {
    /* KAMUS */
    point P, P1;
    float dx, dy;
```

```
    /* ALGORITMA */
    /* Test Baca dan Tulis */
    printf("Masukkan nilai absis dan ordinat : \n");
    BacaPOINT(&P);
    printf("Titik yang dibaca ");
    TulisPOINT(P);
    ...
```



Include file header

# Driver ADT Point

## File: mpoint.c (lanj.)

---

```
...
/* Test IsOrigin */
if (IsOrigin(P)) {
    printf(" adalah titik origin");
}
printf("\n");
/* Test geser */
printf("Geser sejajar sumbu x sebesar = ");
scanf("%f", &dx);
printf("Geser sejajar sumbu y sebesar = ");
scanf("%f", &dy);
printf("Setelah digeser = ");
TulisPOINT(geser(P,dx,dy));
printf("\n");

return 0;
}
```