

# ADT dalam Bahasa C

IF1210 – Algoritma dan Pemrograman 1  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung

# Contoh ADT POINT – Notasi algoritmik

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

```
{*** Konstruktor POINT ***}  
procedure CreatePoint(output p: Point, input x: real, input y: real)  
{ membentuk sebuah Point p 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 menjadi newX }  
procedure setOrdinat (input/output p: Point, input newY: real)  
{ Mengubah nilai komponen Ordinat dari p menjadi newY }
```

# Contoh ADT POINT - Notasi algoritmik (lanj.)

```
{*** Prosedur - Interaksi dengan I/O device, BACA/TULIS ***}  
procedure readPoint (output p: Point)  
{ Membentuk p dari x dan y yang dibaca }  
procedure displayPoint (input p: Point)  
{ Nilai p ditulis ke layar dgn format "(x,y)" }
```

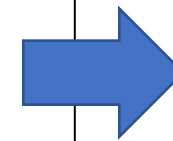
```
{*** Fungsi - Operasi lain terhadap type data ***}  
function move (p1: Point, dx, dy: real) → Point  
{ Menghasilkan Point yang merupakan hasil penggeseran p1 sebesar  
  dx arah sumbu x dan dy arah sumbu y }
```

```
{*** 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 menjadi newX }  
procedure setOrdinat (input/output p: Point, input newY: real)  
{ Mengubah nilai komponen Ordinat dari p menjadi newY }
```



```
#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 Point {
    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 ****/
```

```
void CreatePoint (point* p, 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 readPoint (point * p);
/* Membentuk p dari x dan y yang dibaca dari keyboard */
void displayPoint (point p);
/* Nilai p ditulis ke layar dg format "(X,Y)" */

/**** Fungsi/Operasi lain terhadap point ****/
point move (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 *****/
void CreatePoint (point *p, float x, float y) {
/* Membentuk sebuah point p dari x dan y dengan x sebagai absis dan y sebagai
ordinat */
    /* KAMUS */

    /* ALGORITMA */
    ABSIS(*p) = x;
    ORDINAT(*p) = y;
}
...
```

Contoh Penggunaan  
Selektor



# ADT Point dalam Bahasa C – File point.c (lanj.)

Contoh Penggunaan  
Selektor

...

```
/**** 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));  
}
```

...

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 readPoint (point *p) {  
    /* Membentuk p dari x dan y yang dibaca dari keyboard */  
    /* KAMUS */  
    float x, y;  
    /* ALGORITMA */  
    scanf("%f %f", &x, &y);  
    scanf("%f", &y);  
    CreatePoint(p, x,y);  
}
```

Contoh Penggunaan  
Konstruktor

...

# ADT Point dalam Bahasa C – File point.c (lanj.)

```
...
void displayPoint (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 move (point p, float dx, float dy) {
/* Menghasilkan point yang merupakan hasil penggeseran p */
/* sebesar dx arah sumbu x dan dy arah sumbu y */
/* KAMUS */
    point pt;
/* ALGORITMA */
    CreatePoint(&pt, ABSIS(p)+dx, ORDINAT(p)+dy);
    return (pt);
}
```

# Driver ADT Point – File: main\_point.c

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



Include file header

```
int main () {
    /* KAMUS */
    point p, p1;
    float dx, dy;

    /* ALGORITMA */
    /* Test Baca dan Tulis */
    printf("Masukkan nilai absis dan ordinat: ");
    readPoint(&p);
    printf("Titik yang dibaca ");
    displayPoint(p);
    ...
}
```

# Driver ADT Point – File: main\_point.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 = ");
displayPoint(move(p,dx,dy));
printf("\n");

return 0;
}
```