



Tipe Bentukan

Dasar Pemrograman
Program Studi S1 Informatika
Universitas Diponegoro
Semester Gasal 2020/2021

Outline



Definisi type bentukan

Contoh type bentukan

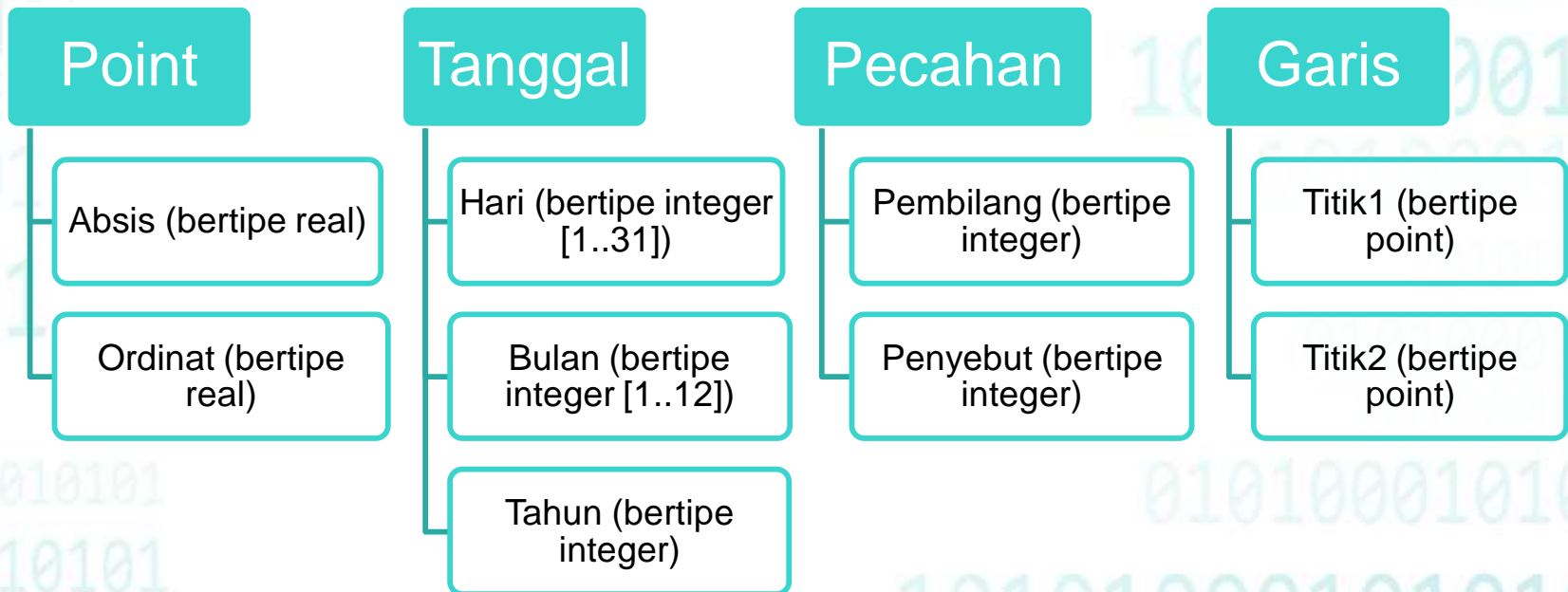
Definisi type bentukan dalam konteks fungsional

Type Bentukan

- Domain dan range dari sebuah fungsi tidak hanya dapat dinyatakan dalam **type dasar** (integer, real, boolean, character, atau string), tetapi dapat pula dinyatakan dalam type yang didefinisikan sendiri.
- **Type bentukan** adalah type yang didefinisikan sendiri, di luar tipe data dasar.
- Type bentukan merupakan **product dari type**.
 - Tersusun atas satu atau lebih type dasar atau type lainnya.
- Disebut juga dengan type komposisi, type terstruktur, atau record.

Type Bentukan (lanj.)

- Type bentukan dapat tersusun atas type dasar atau type lainnya.



Definisi Type

- Nama type
- Komponen pembentuknya beserta domain untuk masing-masing komponen type tersebut.
- Operasi yang dapat dilakukan terhadap type tersebut.
 - Dalam konteks fungsional dinyatakan sebagai fungsi.

Contoh Definisi Nama Type

Contoh definisi type bentukan yang tersusun atas type dasar:

type point : <x: real, y: real>

{<x,y> adalah sebuah point dengan x adalah absis dan y adalah ordinat}

type pecahan : <n: integer, d: integer>0>

{<n,d> adalah sebuah pecahan dengan n adalah numerator (pembilang) dan d adalah denominator (penyebut)}

type date : <d: integer[1..31], m: integer[1..12], y: integer>0>

{ <d,m,y> adalah sebuah tanggal dengan d adalah hari, m adalah bulan dan y adalah tahun}

type time: <h: integer[0..23], m: integer[0..59], s: integer[0..59]>

{ <h,m,s> adalah sebuah waktu dengan h adalah jam, m adalah menit dan s adalah detik}

Contoh Definisi Nama Type (lanj.)

Contoh definisi type bentukan yang tersusun atas tipe lainnya:

type garis : <titik1: point, titik2: point>

{<titik1, titik2> adalah sebuah garis dengan titik1 adalah titik awal dan titik2 adalah titik akhir}

type segiempat : <titik_top: point, titik_bottom: point>

{<titik_top, titik_bottom> membentuk sebuah segiempat dengan titik_top adalah titik atas diagonal dan titik_bottom adalah titik bawah diagonal segiempat}

type datetime : <d: date, t: time>

{ <d,t> membentuk sebuah datetime dengan d adalah tanggal dan t adalah waktu}

Nilai Type

- Nilai type dinyatakan dalam ***tuple*** sesuai dengan komponen pembentuknya.
- Contoh: untuk menyatakan nilai suatu **point** yang didefinisikan oleh ***tuple*** $\langle x: \underline{\text{real}}, y: \underline{\text{real}} \rangle$ digunakan notasi:
 - $\langle 0, 0 \rangle$ artinya $x=0$ dan $y=0$
 - $\langle 2, 1 \rangle$ artinya $x=2$ dan $y=1$
- Contoh: untuk menyatakan nilai suatu **date** yang didefinisikan oleh ***tuple*** $\langle d: \underline{\text{integer}}[1..31], m: \underline{\text{integer}}[1..12], y: \underline{\text{integer}} > 0 \rangle$ digunakan notasi:
 - $\langle 1, 2, 2000 \rangle$ artinya tanggal 1 Februari 2000
 - $\langle 2, 1, 1990 \rangle$ artinya tanggal 2 Januari 1990

Definisi Type dalam Konteks Fungsional

Nama	<ul style="list-style-type: none">• Nama type dan komponennya, hanya akan menjadi definisi.
Selektor	<ul style="list-style-type: none">• Untuk mengakses komponen type menjadi elemen dasar sehingga dapat dioperasikan.• Ditulis definisi dan spesifikasinya dalam bentuk fungsi selektor.• Tidak direalisasikan karena realisasinya sangat tergantung pada ketersediaan bahasa.
Konstruktor	<ul style="list-style-type: none">• Untuk membentuk type komposisi.• Dituliskan definisi dan spesifikasinya sebagai sebuah fungsi• Tidak direalisasikan karena realisasinya sangat tergantung pada ketersediaan bahasa.
Predikat	<ul style="list-style-type: none">• Untuk menentukan karakteristik dan pemeriksaan besaran.
Operator/ Fungsi Lain	<ul style="list-style-type: none">• Fungsi-fungsi lain yang didefinisikan , dibuat spesifikasi beserta realisasinya, berlaku sebagai operator untuk type tersebut.

Definisi Type dalam Konteks Fungsional (lanj.)

- Karena dalam konteks fungsional hanya ada fungsi, semua objek adalah fungsi dan pada akhirnya **ketika realisasi pengertian “type” lenyap**, sehingga kita tidak perlu lagi untuk merealisasikan type.
- Realisasi type dilakukan melalui **konstruktor**.
- Akses ke masing-masing elemen dilakukan melalui **selektor**.
- Predikat dan fungsi/ operator lain menggunakan type sebagai domain atau range dalam fungsi tersebut.
- Definisi type, fungsi konstruktor dan seletor tidak direalisasi dalam notasi fungsional ini → sangat tergantung pada ketersediaan bahasa.



Thank you

Selamat Belajar dan Berlatih !!!