

BAHASA PEMROGRAMAN

TIM PENGAJAR PEMROGRAMAN
Departemen Ilmu Komputer IPB

Pertemuan 2
Paradigma Pemrograman
(**Functional Programming**)

PARADIGMA PEMROGRAMAN

- Paradigma pemrograman adalah bentuk pemecahan masalah mengikuti aliran atau "genre" tertentu dari program dan bahasa.
- Klasifikasi:

Imperative/ Algorithmic	Declarative		Object-Oriented
	Functional Programming	Logic Programming	
Algol Cobol PL/1 Ada C Modula-3	Lisp Haskell ML Miranda APL	Prolog	Smalltalk Simula C++ Java

FUNCTIONAL PROGRAMMING

- Program serba fungsi, artinya setiap persoalan diselesaikan dengan menggunakan fungsi.
- Mulai dikembangkan tahun 1960an, dimotivasi oleh peneliti bidang artificial intelligence, symbolic computation, theorem proving, rule-based system, dan NLP.
- Bahasa fungsional pertama adalah LISP (McCarthy, 1960).
- Memodelkan masalah komputasi sebagai suatu **fungsi matematika**, yang mempunyai input (**domain**) dan hasil atau output (**range**).

FUNGSI DALAM MATEMATIKA

- Fungsi matematika adalah **pemetaan** dari himpunan domain (**daerah fungsi**) ke himpunan lain yang disebut sebagai himpunan range (**wilayah fungsi**).
- Contoh:
 $kuadrat(x) \equiv x * x$ untuk $x \in \mathcal{R}$.
 - Notasi \equiv berarti **didefinisikan sebagai**.
 - $x \in \mathcal{R}$ menunjukkan bahwa daerah fungsi adalah bilangan riil.
- Contoh pemanggilan fungsi:
 $kuadrat(3.0) = 3.0 * 3.0 = 9.0$

EKSPRESI LAMBDA

- **Notasi Lambda** adalah metode menuliskan suatu fungsi matematika tanpa nama. Dirancang oleh Alonzo Church (1941).
- Bentuk fungsi matematika yang ditulis dalam notasi lambda disebut **Ekspresi Lambda**.
- Contoh: $(\lambda x.x * x)$
- Contoh pemanggilan terhadap fungsi lambda: $((\lambda x.x * x)3)$ akan menghasilkan nilai 9.

PEMROGRAMAN FUNGSIONAL

- Pemrograman fungsional **meniru** sejauh mungkin fungsi matematika dalam bentuk **ekspresi lambda**.
- **Pemanggilan fungsi** dalam pemrograman fungsional mirip dengan bentuk **notasi prefix**, disebut dengan **Cambridge Polish** atau **Cambridge Prefix**, yaitu:
(function_name arg₁ arg₂...arg_n)
- Contoh, notasi + adalah fungsi aritmatika penjumlahan, maka 2 ekspresi berikut menghasilkan nilai 12 dan 20.

(+ 5 7)

(+ 3 4 7 6)

DEFINISI FUNGSI

- Mendefinisikan fungsi tanpa nama:
 - Format: `(LAMBDA (parameter) expression)`
 - Contoh: `(LAMBDA (x) (* x x))`
 - Evaluasi: `((LAMBDA (x) (* x x)) 3)`
- Fungsi tanpa parameter:
 - Format: `(DEFINE symbol expression)`
 - Contoh:
`(DEFINE pi 3.14159)`
`(DEFINE two_pi (* 2 pi))`

FUNGSI DENGAN PARAMETER

- **Format:**

```
(DEFINE (function_name parameters)
  (expression)
)
```

- **Contoh:** (DEFINE (kuadrat x) (* x x))

- **Pemanggilan fungsi:** (kuadrat 3) akan menghasilkan nilai 9.

- **Pre-defined function, contoh:**

```
(IF (= n 0) 1 (* n n))
(COND ((< n 5) 1)
      ((< n 10) 2)
      (ELSE 3)
)
```


LATIHAN

Terjemahkan 2 fungsi berikut ke dalam pemrograman fungsional:

$$f(n) = \begin{cases} 1 & n = 0 \\ n * f(n - 1) & n > 1 \end{cases}$$

$$fib(n) = \begin{cases} 0 & n = 0 \\ 1 & n = 1 \\ fib(n - 1) + fib(n - 2) & n > 1 \end{cases}$$

FUNGSI LIST

- **List** adalah himpunan dari 0 atau lebih nilai (elemen/atom)
- Contoh:
(3 5 7 2) dalam matematika ditulis sebagai {3, 5, 7, 2}
- Beberapa fungsi dasar list:
 - (CAR list) → elemen pertama dari list
 - (CDR list) → list selain elemen pertama
 - (CONS atom list) → gabungkan atom ke dalam list
 - (NULL? list) → memeriksa apakah list kosong?

FUNGSI LIST

(CAR ' (a b c)) → a

(CAR ' ((a b) c d)) → (a b)

(CAR ' a) → **error**, karena a bukan list

(CAR ' ()) → **error**

(CDR ' (a b c)) → (b c)

(CDR ' ((a b) c d)) → (c d)

(CDR ' a) → **error**

(CDR ' (a)) → ()

(CDR ' ()) → **error**

(CONS ' a ' ()) → (a)

(CONS ' a ' (b c)) → (a b c)

(CADR ' (a b c)) → (CAR (CDR ' (a b c))) → b

(NULL? ' ()) → #t (**true**)

(NULL? ' (a b c)) → #f (**false**)

LATIHAN

- Perhatikan fungsi berikut:

$$f(list) = \begin{cases} 0 & \text{list kosong} \\ 1 + f(CDR(list)) & \text{selainnya} \end{cases}$$

Apa nilai dari $f(' (2 5 1))$?

Terjemahkan ke dalam pemrograman fungsional.

- Buat fungsi dan program fungsional untuk mendapatkan elemen terakhir dari suatu list. Contoh: $(akhir ' (a b c)) \rightarrow c$.

HOMEWORK

Buat fungsi matematika dan program fungsional untuk masalah berikut (penjelasan detail ada di LMS):

- Menentukan jarak Euclidean dari 2 vektor yang masing-masing dinyatakan dalam list.
- Menghapus elemen list yang sama dan bersebelahan.
- Menghitung banyaknya sifat yang sama dari 2 obyek, dimana setiap sifat obyek dinyatakan dalam bentuk list.

Jawaban diketik menggunakan komputer, simpan dalam format PDF, dan kumpulkan melalui LMS-IPB paling lambat Jumat, 26 Februari 2016 pukul 11am.