



FAKULTAS
ILMU
KOMPUTER

Pemrograman Fungsional 2020

Simply Typed λ Calculus

Rabu, 19 Oktober 2020

Ade Azurat
`ade@cs.ui.ac.id`

Fakultas Ilmu Komputer
Universitas Indonesia



Agenda

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems



λ (Lambda) Calculus

Paparan singkat informal

FunPro 2020

AA

Pengantar

2

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- Kata "**kalkulus**" secara umum berarti sistem baku (formal) untuk aturan inferensi dan aksioma.



λ (Lambda) Calculus

Paparan singkat informal

FunPro 2020

AA

2

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- Kata "**kalkulus**" secara umum berarti sistem baku (formal) untuk aturan inferensi dan aksioma.
- λ *calculus* adalah sebuah sistem baku untuk aturan inferensi dan aksioma yang ditandai dengan symbol λ untuk memodelkan komputasi berfokus/berbasis **ekspresi**.



λ (Lambda) Calculus

Paparan singkat informal

FunPro 2020

AA

2

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- ▶ Kata "**kalkulus**" secara umum berarti sistem baku (formal) untuk aturan inferensi dan aksioma.
- ▶ λ *calculus* adalah sebuah sistem baku untuk aturan inferensi dan aksioma yang ditandai dengan symbol λ untuk memodelkan komputasi berfokus/berbasis **ekspresi**.
- ▶ *The smallest universal programming language*



λ (Lambda) Calculus

Paparan singkat informal

FunPro 2020

AA

2

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- Kata "**kalkulus**" secara umum berarti sistem baku (formal) untuk aturan inferensi dan aksioma.
- λ *calculus* adalah sebuah sistem baku untuk aturan inferensi dan aksioma yang ditandai dengan symbol λ untuk memodelkan komputasi berfokus/berbasis **ekspresi**.
- *The smallest universal programming language*
- Diperkenalkan oleh Alonzo Church tahun 1930-an.



λ (Lambda) Calculus

Paparan singkat informal

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38

- ▶ Kata "**kalkulus**" secara umum berarti sistem baku (formal) untuk aturan inferensi dan aksioma.
- ▶ λ *calculus* adalah sebuah sistem baku untuk aturan inferensi dan aksioma yang ditandai dengan symbol λ untuk memodelkan komputasi berfokus/berbasis **ekspresi**.
- ▶ *The smallest universal programming language*
- ▶ Diperkenalkan oleh Alonzo Church tahun 1930-an.
- ▶ jauh sebelum komputer merupakan barang yang populer.



λ Ekspresi

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

3

38

► Dalam λ *calculus* ada tiga jenis ekspresi, yaitu:

1. variabel (a, b, c, d)
2. aplikasi (*application*)
3. fungsi (*abstraction*)



λ Ekspresi

FunPro 2020

AA

3

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- Dalam λ *calculus* ada tiga jenis ekspresi, yaitu:
 1. variabel (a, b, c, d)
 2. aplikasi (*application*)
 3. fungsi (*abstraction*)
- Secara formal didefinisikan rekursif sebagai berikut:
$$\begin{aligned}\langle \text{ekspresi} \rangle &:= \langle \text{nama} \rangle \mid \langle \text{fungsi} \rangle \mid \langle \text{aplikasi} \rangle \\ \langle \text{fungsi} \rangle &:= \lambda \langle \text{nama} \rangle. \langle \text{ekspresi} \rangle \\ \langle \text{aplikasi} \rangle &:= \langle \text{ekspresi} \rangle \langle \text{ekspresi} \rangle\end{aligned}$$
- nama variabel biasanya dinyatakan oleh huruf kecil ($a, b, c, \dots$)



Asosiatif dalam aplikasi

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

4

1. Ekspresi bisa diberikan tanda kurung.
2. jika E adalah ekspresi, maka (E) adalah ekspresi juga.
3. selain variable dan tanda kurung, simbol lain adalah *lambda* dan tanda titik . (dot).
4. untuk mengurangi tanda kurung, disepakati asosiatif ke kiri yang artinya penulisan sebagai berikut:

$$E_1 E_2 E_3 E_4$$

sama dengan

$$(((E_1 E_2) E_3) E_4)$$



Fungsi

Sebuah λ Ekspresi

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

5

1. Ekspresi:

adalah sebuah fungsi

$\lambda x.y$

38



Fungsi

Sebuah λ Ekspresi

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

5

1. Ekspresi:

$\lambda x.y$

adalah sebuah fungsi

2. variable x setelah simbol λ menyatakan argumen untuk fungsi tersebut.

38



Fungsi

Sebuah λ Ekspresi

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38

1. Ekspresi:

$$\lambda x.y$$

adalah sebuah fungsi

2. variable x setelah simbol λ menyatakan argumen untuk fungsi tersebut.
3. variable y setelah simbol $.$ (*dot*) adalah isi (*body*) dari fungsi tersebut.



Fungsi dalam fungsi

Sebuah λ Ekspresi

FunPro 2020

AA

6

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- di dalam isi sebuah fungsi, bisa berupa ekspresi lagi yang ekspresinya bisa berbentuk salah satu dari tiga bentuk tersebut (variable, fungsi, application)



Fungsi dalam fungsi

Sebuah λ Ekspresi

FunPro 2020

AA

6

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- ▶ di dalam isi sebuah fungsi, bisa berupa ekspresi lagi yang ekspresinya bisa berbentuk salah satu dari tiga bentuk tersebut (variable, fungsi, application)

- ▶ Ekspresi:

$$\lambda x. \lambda y. z$$

adalah sebuah fungsi.



Fungsi dalam fungsi

Sebuah λ Ekspresi

FunPro 2020

AA

6

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- ▶ di dalam isi sebuah fungsi, bisa berupa ekspresi lagi yang ekspresinya bisa berbentuk salah satu dari tiga bentuk tersebut (variable, fungsi, application)

- ▶ Ekspresi:

$$\lambda x. \lambda y. z$$

adalah sebuah fungsi.

- ▶ Fungsi tersebut diatas memiliki argumen x yang di dalam *isi* fungsi tersebut, terdapat fungsi lain. Fungsi yang berada di dalam memiliki argument y dan isinya adalah z



Terminolog

Fungsi, λ ekspresi, dan aplikasi fungsi

FunPro 2020

AA

7

► Ekspresi:

$\lambda x.x$

adalah sebuah fungsi identitas

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer

Universitas Indonesia

38



Terminolog

Fungsi, λ ekspresi, dan aplikasi fungsi

FunPro 2020

AA

7

► Ekspresi:

$\lambda x.x$

adalah sebuah fungsi identitas

► Sebuah ekspresi bisa di-**aplikasi**-kan kepada sebuah ekspresi.

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38



Terminolog

Fungsi, λ ekspresi, dan aplikasi fungsi

FunPro 2020

AA

7

► Ekspresi:

$\lambda x.x$

adalah sebuah fungsi identitas

► Sebuah ekspresi bisa di-**aplikasi**-kan kepada sebuah ekspresi.

► Ekspresi:

$(\lambda x.x)y$

adalah aplikasi ekspresi y (yang dalam hal ini adalah variable) kepada fungsi identitas tersebut.

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38



Terminolog

Fungsi, λ ekspresi, dan aplikasi fungsi

FunPro 2020

AA

7

► Ekspresi:

$\lambda x.x$

adalah sebuah fungsi identitas

► Sebuah ekspresi bisa di-**aplikasi**-kan kepada sebuah ekspresi.

► Ekspresi:

$(\lambda x.x)y$

adalah aplikasi ekspresi y (yang dalam hal ini adalah variable) kepada fungsi identitas tersebut.

► **Aplikasi fungsi** adalah proses meng-substitusi kehadiran variable pada argument x dengan ekspresi y yang artinya setiap kemunculan variable x dalam isi fungsi tersebut akan digantikan oleh variable y .

$(\lambda x.x)y \equiv y$

38



β (beta) Reduction *λ calculus*

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

8

- Secara umum aplikasi

$$(\lambda v. E_1) E_2$$

dievaluasi sebagai

$$E_1[v := E_2]$$

- dibaca sebagai kehadiran variabel v didalam E_1 digantikan dengan E_2 .
- Aturan ini secara formal disebut β (**beta**) **reduction**
- Contoh:

$$\begin{aligned} & (\lambda x. (\lambda y. x)) z \\ & \equiv (\lambda y. x)[x := z] \\ & \equiv (\lambda y. z) \end{aligned}$$



variabel bebas dan variable terikat

λ calculus

FunPro 2020

AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

9

Evaluasi dan
Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

- ▶ dalam λ calculus nama variabel adalah berlaku local didalam definisi tersebut.
- ▶ dalam ekspresi

$$\lambda y.y$$

variabel y dikatakan **terikat**, sementara variable lain yang tidak didahului oleh simbol λ dikatakan sebagai variabel **bebas**

- ▶ Misalkan pada ekspresi:

$$\lambda x.xy$$

maka variabel x terikat dan variabel y dinyatakan bebas.



variabel bebas dan variable terikat

λ calculus

FunPro 2020

AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

9

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan
Beta Reduction
Type Systems

- ▶ dalam λ calculus nama variabel adalah berlaku local didalam definisi tersebut.
- ▶ dalam ekspresi

$$\lambda y. y$$

variabel y dikatakan **terikat**, sementara variable lain yang tidak didahului oleh simbol λ dikatakan sebagai variabel **bebas**

- ▶ Misalkan pada ekspresi:

$$\lambda x. xy$$

maka variabel x terikat dan variabel y dinyatakan bebas.

- ▶ pada contoh berikut:

$$(\lambda x. x)(\lambda y. yx)$$

perhatikan bahwa variable x di ekspresi sebelah kiri terikat, tapi variabel x di ekspresi sebelah kanan adalah bebas. Kedua variable x tersebut menyatakan hal yang berbeda.



observasi: Fungsi Identitas

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

10

► fungsi identitas

$\lambda x.x$

► Contoh evaluasi:

$(\lambda x.x)z$
 $\equiv x[x := z]$
 $\equiv z$

38



observasi: Fungsi Identitas

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

10

- fungsi identitas

$\lambda x.x$

- Contoh evaluasi:

$(\lambda x.x)z$
 $\equiv x[x := z]$
 $\equiv z$

- fungsi

$\lambda y.y$

adalah juga fungsi identitas

- Contoh evaluasi:

$(\lambda y.y)z$
 $\equiv y[y := z]$
 $\equiv z$

38



observasi: Fungsi Identitas

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

10

- fungsi identitas

$$\lambda x.x$$

- Contoh evaluasi:

$$\begin{aligned} & (\lambda x.x)z \\ & \equiv x[x := z] \\ & \equiv z \end{aligned}$$

- fungsi

$$\lambda y.y$$

adalah juga fungsi identitas

- Contoh evaluasi:

$$\begin{aligned} & (\lambda y.y)z \\ & \equiv y[y := z] \\ & \equiv z \end{aligned}$$

secara umum variabel terikat didalam sebuah ekspresi bisa diganti dengan variabel baru lain, asalkan variable baru tersebut tidak muncul bebas dialam ekspresi tersebut.

38



α (alpha) Conversion *λ calculus*

FunPro 2020

AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

11

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan
Beta Reduction
Type Systems

- secara general setiap variable bisa digantikan dengan variabel lain dan tetap menyatakan ekspresi yang sama (asalkan dilakukan secara konsisten).

- Contoh evaluasi:

$$\begin{aligned} &(\lambda x.x)z \\ &\equiv x[x := z] \\ &\equiv z \end{aligned}$$

- fungsi

$$\lambda y.y$$

adalah juga fungsi identitas

- Contoh evaluasi:

$$\begin{aligned} &(\lambda y.y)z \\ &\equiv y[y := z] \\ &\equiv z \end{aligned}$$

- secara umum variabel didalam sebuah ekspresi bisa secara konsisten diganti

38



Fungsi tidak punya nama

λ calculus

- Dalam λ calculus , setiap fungsi tidak diberi nama.
- Dalam mengevaluasinya kita perlu menuliskan keseluruhan fungsi.

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

12

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38



Fungsi tidak punya nama

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

12

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- ▶ Dalam λ calculus , setiap fungsi tidak diberi nama.
- ▶ Dalam mengevaluasinya kita perlu menuliskan keseluruhan fungsi.
- ▶ Misalkan fungsi identitas nyatakan equivalen dengan ekspresi $I \equiv (\lambda x.x)$
- ▶ Ekspresi I merupakan sinonim dengan fungsi identitas tersebut.



Fungsi tidak punya nama

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

12

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- ▶ Dalam λ calculus, setiap fungsi tidak diberi nama.
- ▶ Dalam mengevaluasinya kita perlu menuliskan keseluruhan fungsi.
- ▶ Misalkan fungsi identitas nyatakan equivalen dengan ekspresi $I \equiv (\lambda x.x)$
- ▶ Ekspresi I merupakan sinonim dengan fungsi identitas tersebut.
- ▶ Fungsi identitas I bisa kita aplikasikan dengan fungsi itu sendiri:

$$II \equiv (\lambda x.x)(\lambda x.x)$$

- ▶ sebagaimana observasi sebelumnya, untuk menghindari kebingungan dengan adanya variabel yang sama tapi mungkin punya makna berbeda, kita dapat mengubah variable di salah salah ekspresi dengan menerapkan α – conversion menjadi

$$II \equiv (\lambda x.x)(\lambda z.z)$$



Fungsi tidak punya nama

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

12

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- ▶ Dalam λ calculus, setiap fungsi tidak diberi nama.
- ▶ Dalam mengevaluasinya kita perlu menuliskan keseluruhan fungsi.
- ▶ Misalkan fungsi identitas nyatakan equivalen dengan ekspresi $I \equiv (\lambda x.x)$
- ▶ Ekspresi I merupakan sinonim dengan fungsi identitas tersebut.
- ▶ Fungsi identitas I bisa kita aplikasikan dengan fungsi itu sendiri:

$$II \equiv (\lambda x.x)(\lambda x.x)$$

- ▶ sebagaimana observasi sebelumnya, untuk menghindari kebingungan dengan adanya variabel yang sama tapi mungkin punya makna berbeda, kita dapat mengubah variable di salah salah ekspresi dengan menerapkan α – conversion menjadi

$$II \equiv (\lambda x.x)(\lambda z.z)$$

- ▶ Evaluasi ekspresi tersebut:

$$\begin{aligned} II &\equiv (\lambda x.x)(\lambda z.z) \\ &\equiv x[x := (\lambda z.z)] \\ &\equiv (\lambda z.z) \\ &\equiv I \end{aligned}$$

38



Tidak mencampur variabel bebas dan variabel terikat

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

13

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

38

- ▶ Ketika melakukan substitusi, kita perlu hati-hati agar tidak mencampur antara variable bebas dengan variabel terikat.
- ▶ Misalkan ekspresi berikut ini

$$(\lambda x. (\lambda y. xy))y$$



Tidak mencampur variabel bebas dan variabel terikat

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

13

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- ▶ Ketika melakukan substitusi, kita perlu hati-hati agar tidak mencampur antara variable bebas dengan variabel terikat.
- ▶ Misalkan ekspresi berikut ini

$$(\lambda x.(\lambda y.xy))y$$

- ▶ Perhatikan variabel y pada xy adalah terikat, sementara variabel y yang paling kanan adalah bebas.



Tidak mencampur variabel bebas dan variabel terikat

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

13

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- ▶ Ketika melakukan substitusi, kita perlu hati-hati agar tidak mencampur antara variable bebas dengan variabel terikat.
- ▶ Misalkan ekspresi berikut ini

$$(\lambda x.(\lambda y.xy))y$$

- ▶ Perhatikan variabel y pada xy adalah terikat, sementara variabel y yang paling kanan adalah bebas.
- ▶ Substitusi yang keliru akan menghasilkan:

$$\begin{aligned} &(\lambda x.(\lambda y.xy))y \\ &\equiv (\lambda y.xy)[x := y] \\ &\equiv (\lambda y.yy) \end{aligned}$$

- ▶ kedua variabel y yang mungkin berbeda, akhirnya kedua jadi terikat dan harus sama.



Tidak mencampur variabel bebas dan variabel terikat

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

13

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- ▶ Ketika melakukan substitusi, kita perlu hati-hati agar tidak mencampur antara variabel bebas dengan variabel terikat.
- ▶ Misalkan ekspresi berikut ini

$$(\lambda x.(\lambda y.xy))y$$

- ▶ Perhatikan variabel y pada xy adalah terikat, sementara variabel y yang paling kanan adalah bebas.
- ▶ Substitusi yang keliru akan menghasilkan:

$$\begin{aligned} &(\lambda x.(\lambda y.xy))y \\ &\equiv (\lambda y.xy)[x := y] \\ &\equiv (\lambda y.yy) \end{aligned}$$

- ▶ kedua variabel y yang mungkin berbeda, akhirnya kedua jadi terikat dan harus sama.
- ▶ Sebelum di evaluasi, seharusnya bisa diganti dulu variabel terikat nya, yaitu:

$$\begin{aligned} &(\lambda x.(\lambda y.xy))y \\ &\equiv (\lambda x.(\lambda t.xt))y \\ &\equiv (\lambda t.xt)[x := y] \\ &\equiv (\lambda t.yt) \end{aligned}$$

38



Representasi bilangan cacah

λ calculus : Church Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

14

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- Sebuah bahasa pemrograman perlu dapat merepresentasikan bilangan



Representasi bilangan cacah

λ calculus : Church Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

14

- Sebuah bahasa pemrograman perlu dapat merepresentasikan bilangan
- Sebelumnya kita telah pelajari bagaimana representasi bilangan dinyatakan secara simbol (gambar),

38



Representasi bilangan cacah

λ calculus : Church Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

14

- Sebuah bahasa pemrograman perlu dapat merepresentasikan bilangan
- Sebelumnya kita telah pelajari bagaimana representasi bilangan dinyatakan secara simbol (gambar),
- kemudian ketika kuliah kita mempelajari bahwa bilangan bisa dinyatakan secara binary berdasarkan signal listrik.

38



Representasi bilangan cacah

λ calculus : Church Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

14

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38

- ▶ Sebuah bahasa pemrograman perlu dapat merepresentasikan bilangan
- ▶ Sebelumnya kita telah pelajari bagaimana representasi bilangan dinyatakan secara simbol (gambar),
- ▶ kemudian ketika kuliah kita mempelajari bahwa bilangan bisa dinyatakan secara binary berdasarkan signal listrik.
- ▶ Pada kali ini kita akan pelajari bagaimana **bilangan dinyatakan sebagai fungsi** dalam λ calculus



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

15

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

► $0 \equiv \lambda s.(\lambda z.z)$

Fakultas Ilmu Komputer
Universitas Indonesia

38



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

15

- ▶ $0 \equiv \lambda s.(\lambda z.z)$
- ▶ Bila ada symbol λ bersarang biasa nya dipersingkat penulisnya menjadi:
 $0 \equiv \lambda sz.z$



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

15

- ▶ $0 \equiv \lambda s.(\lambda z.z)$
- ▶ Bila ada symbol λ bersarang biasa nya dipersingkat penulisnya menjadi:
 $0 \equiv \lambda sz.z$
- ▶ $1 \equiv \lambda sz.s(z)$

38



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

15

- ▶ $0 \equiv \lambda s.(\lambda z.z)$
- ▶ Bila ada symbol λ bersarang biasa nya dipersingkat penulisnya menjadi:
 $0 \equiv \lambda sz.z$
- ▶ $1 \equiv \lambda sz.s(z)$
- ▶ $2 \equiv \lambda sz.s(s(z))$

38



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

15

- ▶ $0 \equiv \lambda s.(\lambda z.z)$
- ▶ Bila ada symbol λ bersarang biasa nya dipersingkat penulisnya menjadi:
 $0 \equiv \lambda sz.z$
- ▶ $1 \equiv \lambda sz.s(z)$
- ▶ $2 \equiv \lambda sz.s(s(z))$
- ▶ $3 \equiv \lambda sz.s(s(s(z)))$
- ▶ dan seterusnya

38



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

16

- Di λ calculus ada sebuah fungsi yang disebut fungsi *successor* dengan simbol S .

38



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

16

- Di λ calculus ada sebuah fungsi yang disebut fungsi *successor* dengan simbol S .
- Fungsi tersebut dapat memodelkan bilangan secara unary: $1 = S(0)$, $2 = S(S(0))$ dan seterusnya.

38



Representasi bilangan cacah

λ calculus : Church's Numerals

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

16

- ▶ Di λ calculus ada sebuah fungsi yang disebut fungsi *successor* dengan simbol S .
- ▶ Fungsi tersebut dapat memodelkan bilangan secara unary: $1 = S(0)$, $2 = S(S(0))$ dan seterusnya.
- ▶ Definisinya:

$$S \equiv \lambda wyx. y(wyx)$$

38



Penjumlahan

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

17

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38

► Misalkan kita ingin menjumlahkan

$$2 + 3$$



Penjumlahan

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

17

- Misalkan kita ingin menjumlahkan

$$2 + 3$$

- dalam λ calculus dapat dimodelkan dengan

$$2S3$$

38



Penjumlahan

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

17

- Misalkan kita ingin menjumlahkan

$$2 + 3$$

- dalam λ calculus dapat dimodelkan dengan

$$2S3$$

- $2S3 \equiv (\lambda sz.s(s(z)))(\lambda wyx.y(wyx))(\lambda uv.u(u(u(v))))$

38



Penjumlahan

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlah

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

17

- ▶ Misalkan kita ingin menjumlahkan

$$2 + 3$$

- ▶ dalam λ calculus dapat dimodelkan dengan

$$2S3$$

- ▶ $2S3 \equiv (\lambda sz.s(s(z)))(\lambda wyx.y(wyx))(\lambda uv.u(u(u(v))))$
- ▶ apakah bila dievaluasi benar hasilnya 5?

38



Penjumlahan

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

17

Penjumlah

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38

- ▶ Misalkan kita ingin menjumlahkan

$$2 + 3$$

- ▶ dalam λ calculus dapat dimodelkan dengan

$$2S3$$

- ▶ $2S3 \equiv (\lambda sz.s(s(z)))(\lambda wyx.y(wyx))(\lambda uv.u(u(u(v))))$
- ▶ apakah bila dievaluasi benar hasilnya 5?
- ▶ Mari coba dulu yang lebih sederhana dengan

$$0 + 1$$

$$0S1$$

apakah benar hasilnya 1?



Perkalian

λ calculus

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

18

Fakultas Ilmu Komputer
Universitas Indonesia

38



Perkalian

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

- Perkalian

$$0x1$$

dimodelkan dengan

$$(\lambda xyz.x(yz))01$$

- Pada penulisan kali ini kita perkenalkan 'literal 0 dan 1' yang kelak akan dikembalikan kepada definisi representasinya di λ calculus

18

38



Perkalian

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

- Perkalian

$$0x1$$

dimodelkan dengan

$$(\lambda xyz.x(yz))01$$

- Pada penulisan kali ini kita perkenalkan 'literal 0 dan 1' yang kelak akan dikembalikan kepada definisi representasinya di λ calculus

►

$$0 * 1$$

$$\equiv (\lambda xyz.x(yz))01$$

$$\equiv (\lambda yz.0(yz))1$$

representasi dari $0 * 1$
aplikasi $0, x := 0$

18

38



Perkalian

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

- Perkalian

$$0x1$$

dimodelkan dengan

$$(\lambda xyz.x(yz))01$$

- Pada penulisan kali ini kita perkenalkan 'literal 0 dan 1' yang kelak akan dikembalikan kepada definisi representasinya di λ calculus

►

$$0 * 1$$

$$\equiv (\lambda xyz.x(yz))01$$

$$\equiv (\lambda yz.0(yz))1$$

$$\equiv (\lambda z.0(1z))$$

representasi dari $0 * 1$

aplikasi $0, x := 0$

aplikasi $1, y := 1$

18

38



Perkalian

λ calculus

FunPro 2020
AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan

Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan
Beta Reduction
Type Systems

18

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

- Perkalian

$$0x1$$

dimodelkan dengan

$$(\lambda xyz.x(yz))01$$

- Pada penulisan kali ini kita perkenalkan 'literal 0 dan 1' yang kelak akan dikembalikan kepada definisi representasinya di λ calculus

►

$$0 * 1$$

$$\equiv (\lambda xyz.x(yz))01$$

$$\equiv (\lambda yz.0(yz))1$$

$$\equiv (\lambda z.0(1z))$$

$$\equiv (\lambda z.(\lambda s.(\lambda t.t))(1z)) \quad \text{membuka definisi literal } 0 \equiv (\lambda s.(\lambda t.t))$$

representasi dari $0 * 1$

aplikasi $0, x := 0$

aplikasi $1, y := 1$

38



Perkalian

λ calculus

FunPro 2020
AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan

Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan
Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

- Perkalian

$$0x1$$

dimodelkan dengan

$$(\lambda xyz.x(yz))01$$

- Pada penulisan kali ini kita perkenalkan 'literal 0 dan 1' yang kelak akan dikembalikan kepada definisi representasinya di λ calculus

►

$$0 * 1$$

$$\equiv (\lambda xyz.x(yz))01$$

$$\equiv (\lambda yz.0(yz))1$$

$$\equiv (\lambda z.0(1z))$$

$$\equiv (\lambda z.(\lambda s.(\lambda t.t))(1z))$$

$$\equiv (\lambda z.(\lambda s.(\lambda t.t))(1z))$$

representasi dari $0 * 1$

aplikasi $0, x := 0$

aplikasi $1, y := 1$

membuka definisi literal $0 \equiv (\lambda s.(\lambda t.t))$

aplikasi $(1z), s := (1z)$

18

38



Perkalian

λ calculus

FunPro 2020
AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan
Perkalian

Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan
Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

- Perkalian adalah pengulangan penjumlahan dan dapat dimodelkan dengan

$$\lambda xyz.x(yz)$$

- Perkalian

$$0x1$$

dimodelkan dengan

$$(\lambda xyz.x(yz))01$$

- Pada penulisan kali ini kita perkenalkan 'literal 0 dan 1' yang kelak akan dikembalikan kepada definisi representasinya di λ calculus

►

$$0 * 1$$

$$\equiv (\lambda xyz.x(yz))01$$

$$\equiv (\lambda yz.0(yz))1$$

$$\equiv (\lambda z.0(1z))$$

$$\equiv (\lambda z.(\lambda s.(\lambda t.t))(1z))$$

$$\equiv (\lambda z.(\lambda s.(\lambda t.t))(1z))$$

$$\equiv (\lambda z.(\lambda t.t))$$

$$\equiv 0$$

representasi dari $0 * 1$

aplikasi $0, x := 0$

aplikasi $1, y := 1$

membuka definisi literal $0 \equiv (\lambda s.(\lambda t.t))$

aplikasi $(1z), s := (1z)$

definisi 0

- Bagaimana dengan perkalian $1 * 2$ (Kita akan berlatih dan kerjakan bersama)?



Kondisi Bersyarat (If-then-else)

λ calculus : Church Boolean

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

T (true) dinyatakan dengan $\lambda xy.x$

F (false) dinyatakan dengan $\lambda xy.y$

19

Fakultas Ilmu Komputer
Universitas Indonesia

38



Kondisi Bersyarat (If-then-else)

λ calculus : Church Boolean

T (true) dinyatakan dengan $\lambda xy.x$

F (false) dinyatakan dengan $\lambda xy.y$

Contoh: Bagaimana menyatakan **if P then E_1 else E_2**

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

19

Fakultas Ilmu Komputer
Universitas Indonesia

38



Kondisi Bersyarat (If-then-else)

λ calculus : Church Boolean

T (true) dinyatakan dengan $\lambda xy.x$

F (false) dinyatakan dengan $\lambda xy.y$

Contoh: Bagaimana menyatakan **if P then E_1 else E_2**

- ▶ P adalah sebuah propotional yang hanya dapat berisi benar atau salah.
- ▶ Dengan kata lain, P dinyatakan sebuah variable yang bisa berisi literal T atau F .

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

19

38

Fakultas Ilmu Komputer
Universitas Indonesia



Kondisi Bersyarat (If-then-else)

λ calculus : Church Boolean

T (true) dinyatakan dengan $\lambda xy.x$

F (false) dinyatakan dengan $\lambda xy.y$

Contoh: Bagaimana menyatakan **if P then E_1 else E_2**

- ▶ P adalah sebuah propotional yang hanya dapat berisi benar atau salah.
- ▶ Dengan kata lain, P dinyatakan sebuah variable yang bisa berisi literal T atau F .
- ▶ sehingga pernyataan tersebut bisa dituliskan dalam λ calculus dengan

$$PE_1E_2$$

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

19

Fakultas Ilmu Komputer
Universitas Indonesia

38



Kondisi Bersyarat (If-then-else)

λ calculus : Church Boolean

T (true) dinyatakan dengan $\lambda xy.x$

F (false) dinyatakan dengan $\lambda xy.y$

Contoh: Bagaimana menyatakan **if P then E_1 else E_2**

- ▶ P adalah sebuah propotional yang hanya dapat berisi benar atau salah.
- ▶ Dengan kata lain, P dinyatakan sebuah variable yang bisa berisi literal T atau F .
- ▶ sehingga pernyataan tersebut bisa dituliskan dalam λ calculus dengan

$$PE_1E_2$$

- ▶ Bila P adalah benar (T) maka

$$\begin{aligned} PE_1E_2 & \quad P \text{ adalah true} \\ \equiv (\lambda xy.x)E_1E_2 & \quad \beta - \text{reduction} \\ \equiv E_1 \end{aligned}$$

- ▶ Bila P adalah false (F) maka

$$\begin{aligned} PE_1E_2 & \quad P \text{ adalah false} \\ \equiv (\lambda xy.y)E_1E_2 & \quad \beta - \text{reduction} \\ \equiv E_2 \end{aligned}$$



Operasi Logika

λ calculus : Church's Boolean

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Conjunction (and) ($\wedge$) dinyatakan dengan

$$\wedge \equiv \lambda xy.xy(\lambda uv.v) \equiv \lambda xy.xyF$$

Disjunction (or) ($\vee$) dinyatakan dengan

$$\vee \equiv \lambda xy.x(\lambda uv.u)y \equiv \lambda xy.xTy$$

Negation (not) ($\neg$) dinyatakan dengan

$$\neg \equiv \lambda x.x(\lambda uv.v)(\lambda ab.a) \equiv \lambda x.xFT$$

20

38



Operasi Logika

λ calculus : Church's Boolean

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

20

38

Conjunction (and) ($\wedge$) dinyatakan dengan

$$\wedge \equiv \lambda xy.xy(\lambda uv.v) \equiv \lambda xy.xyF$$

Disjunction (or) ($\vee$) dinyatakan dengan

$$\vee \equiv \lambda xy.x(\lambda uv.u)y \equiv \lambda xy.xTy$$

Negation (not) ($\neg$) dinyatakan dengan

$$\neg \equiv \lambda x.x(\lambda uv.v)(\lambda ab.a) \equiv \lambda x.xFT$$

► Contoh: $\neg T \equiv F$.

$$\begin{aligned} \neg T & \quad \text{definisi } \neg \\ \equiv (\lambda x.xFT)T & \quad \beta - \text{reduction} \\ \equiv TFT & \quad \text{definisi } T \\ \equiv (\lambda xy.x)FT & \quad \beta - \text{reduction} \\ \equiv F & \end{aligned}$$



Test zero

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

21

Dalam beberapa bahasa pemrograman akan sangat bermanfaat bila memiliki fungsi yang menguji apakah suatu variabel bernilai 0 atau tidak. Untuk kebutuhan tersebut misalkan kita definisikan

$$Z \equiv \lambda x.xF \neg F$$

Untuk memahami cara kerjanya, perhatikan bahwa:

$$0fa \equiv (\lambda sz.z)fa \equiv a$$

yang artinya fungsi f , diaplikasikan dengan a sebanyak 0 kali sehingga tetap a .

38



Test zero

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

21

38

Dalam beberapa bahasa pemrograman akan sangat bermanfaat bila memiliki fungsi yang menguji apakah suatu variabel bernilai 0 atau tidak. Untuk kebutuhan tersebut misalkan kita definisikan

$$Z \equiv \lambda x. xF \neg F$$

Untuk memahami cara kerjanya, perhatikan bahwa:

$$0fa \equiv (\lambda sz. z)fa \equiv a$$

yang artinya fungsi f , diaplikasikan dengan a sebanyak 0 kali sehingga tetap a . Perhatikan juga bahwa bila fungsi F diaplikasikan dengan apapun, maka akan menghasilkan fungsi identitas I .

$$Fa \equiv (\lambda xy. y)a \equiv \lambda y. y \equiv I$$

Sekarang kita perhatikan bagaimana cara kerja fungsi Z ini. Perhatikan fungsi Z diaplikasi dengan 0.

$$\begin{aligned} Z0 &\equiv (\lambda x. xF \neg F)0 \\ &\equiv 0F \neg F \\ &\equiv \neg F \\ &\equiv T \end{aligned}$$

definisi Z dan β - reduction
karena F diaplikasi dengan $\neg F$ sebanyak 0 kali



Test zero

λ calculus

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan
Beta Reduction
Type Systems

Representasi *church numeral*, menyatakan bilangan sebagai enumerasi yang merupakan fungsi menerima fungsi dan argumen kemudian menerapkan fungsi tersebut terus menerus sebanyak enumerasinya.
Ingat bahwa bila fungsi F diaplikasikan dengan apapun, maka akan menghasilkan fungsi identitas I .

$$Fa \equiv (\lambda xy.y)a \equiv \lambda y.y \equiv I$$

22 Perhatikan fungsi Z diaplikasi dengan bilangan bukan 0, misalkan N .

$$\begin{aligned} ZN &\equiv (\lambda x.xF\neg F)N && \text{definisi } Z \text{ dan } \beta - \text{reduction} \\ &\equiv NF\neg F && \text{karena } F \text{ diaplikasi dengan } \neg F \text{ sebanyak } N \text{ kali, hasilnya } I \\ &\equiv IF \\ &\equiv F \end{aligned}$$



Predecessor (Nilai sebelumnya)

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

Untuk menghitung nilai sebelumnya, digunakan representasi pair ($n, n - 1$) elemen kedua dari pair dinyatakan sebagai *predecessor*. Sebuah *pair*(a, b) dinyatakan dalam λ calculus dengan

$$(a, b) \equiv \lambda z. zab$$

Untuk mengambil elemen pertama, fungsi pair tersebut di aplikasikan dengan T .

$$first(a, b) \equiv (\lambda z. zab) T \equiv Tab \equiv a$$

dan untuk mengambil elemen kedua, fungsi pair tersebut diaplikasikan dengan F .

$$second(a, b) \equiv (\lambda z. zab) F \equiv Fab \equiv b$$

Untuk dapat menggenerate pair ke- n , didefinisikan fungsi generator:

$$\Phi \equiv (\lambda pz. z(S(pT))(pT))$$

Fungsi predecessor didefinisikan dengan sebelumnya menggenerate pair ke- n , kemudian mengambil elemen kedua. Generator akan menerapkan sebanyak n kali dari inialisasi awal yaitu $\lambda z. z00$. Perhatikan bahwa kita memodelkan sehingga dianggap predecessor dari 0 adalah 0. Dengan acuan tersebut didefinisikan:

$$P \equiv (\lambda n. n \Phi (\lambda z. z00) F)$$

23

38



Equality dan inequalities

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

24

38

Dengan memanfaatkan definisi predecessor P kita bisa definisikan "predikat lebih besar sama dengan" ($\geq$) kita simbolkan dengan G sebagai berikut:

$$G \equiv (\lambda xy. Z(xPy))$$

Ini artinya kita memeriksa, apakah bila fungsi predecessor P diaplikasikan dengan y sebanyak x kali akan menghasilkan 0 atau tidak. Bila x lebih besar atau sama dengan y maka hasilnya akan 0, karena y akan dikurangi satu kali terus menerus sebanyak x kali. Berlaku juga sebaliknya. Bila hasilnya 0, maka fungsi penguji Z akan memberikan jawaban T (true).

Untuk mendefinisikan persamaan (*equality*) dari bilangan didefinisikan E :

$$E \equiv (\lambda xy. \wedge (Z(xPy))(Z(yPx)))$$

yang menyatakan bahwa bila $x \geq y$ dan $y \geq x$ maka $x = y$



Rekursif

λ calculus

FunPro 2020

AA

Pengantar
Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan
Substitusi

Representasi
Church Numerals
Penjumlahan
Perkalian

Church Boolean
True False
Operasi Logika
Equality dan inequalities

Rekursif

25

Type System
Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

38

Definisi rekursif dimodelkan dalam λ calculus dengan fungsi yang disebut Y dan fungsi tersebut me-regenerate fungsi kembali. Didefinisikan sebagai berikut:

$$Y \equiv (\lambda y. (\lambda x. y(xx)) (\lambda x. y(xx)))$$

Bagaimana penerapan-nya? Misalkan ada sebuah fungsi rekursif R .

$$\begin{aligned} YR &\equiv ((\lambda x. R(xx)) (\lambda x. R(xx))) && \text{definisi} \\ &\equiv R((\lambda x. R(xx)) (\lambda x. R(xx))) && \beta - \text{reduction} \\ &\equiv R(YR) && \text{rekursif} \end{aligned}$$

Dengan kata lain, bila fungsi Y diaplikasikan dengan sebuah fungsi R , maka akan menghasilkan pemanggilan rekursi dari fungsi R tersebut.

Contoh:

$$\sum_{i=0}^n i = n + \sum_{i=0}^{n-1} i$$



Rekursif

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Contoh:

$$Y \equiv (\lambda y. (\lambda x. y(xx)) (\lambda x. y(xx)))$$

$$\sum_{i=0}^n i = n + \sum_{i=0}^{n-1} i$$

26

Fakultas Ilmu Komputer
Universitas Indonesia

38



Rekursif

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

26

38

$$Y \equiv (\lambda y. (\lambda x. y(xx)) (\lambda x. y(xx)))$$

Contoh:

$$\sum_{i=0}^n i = n + \sum_{i=0}^{n-1} i$$

Fungsi rekursif tersebut diberi nama R dan dimodelkan dalam λ *calculus* menjadi:

$$R \equiv (\lambda rn. Zn0(nS(r(Pn))))$$



Rekursif

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

26

38

$$Y \equiv (\lambda y. (\lambda x. y(xx)) (\lambda x. y(xx)))$$

Contoh:

$$\sum_{i=0}^n i = n + \sum_{i=0}^{n-1} i$$

Fungsi rekursif tersebut diberi nama R dan dimodelkan dalam λ calculus menjadi:

$$R \equiv (\lambda rn. Zn0(nS(r(Pn))))$$

Misalkan dengan parameter $n = 3$, penerapan rekursifnya menjadi:

$$YR3 = R(YR)3 = Z30(3S(YR(P3)))$$



Rekursif

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

26

38

$$Y \equiv (\lambda y. (\lambda x. y(xx)) (\lambda x. y(xx)))$$

Contoh:

$$\sum_{i=0}^n i = n + \sum_{i=0}^{n-1} i$$

Fungsi rekursif tersebut diberi nama R dan dimodelkan dalam λ calculus menjadi:

$$R \equiv (\lambda rn. Zn0(nS(r(Pn))))$$

Misalkan dengan parameter $n = 3$, penerapan rekursifnya menjadi:

$$YR3 = R(YR)3 = Z30(3S(YR(P3)))$$

Karena 3 tidak sama dengan 0, maka menjadi:

$$3S(YR2)$$



Rekursif

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

$$Y \equiv (\lambda y. (\lambda x. y(xx)) (\lambda x. y(xx)))$$

Contoh:

$$\sum_{i=0}^n i = n + \sum_{i=0}^{n-1} i$$

Fungsi rekursif tersebut diberi nama R dan dimodelkan dalam λ calculus menjadi:

$$R \equiv (\lambda rn. Zn0(nS(r(Pn))))$$

Misalkan dengan parameter $n = 3$, penerapan rekursifnya menjadi:

$$YR3 = R(YR)3 = Z30(3S(YR(P3)))$$

Karena 3 tidak sama dengan 0, maka menjadi:

$$3S(YR2)$$

Bila kembali diterapkan definisi Y , maka akan didapat:

$$3S(YR2) \equiv 3S(2S(YR1)) \equiv 3S(2S(1S0)) \equiv 6$$

26

38



Motivasi untuk Sistem Tipe (Simply Typed Lambda Calculus)

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

27

Sebuah Paradox
Perhatikan kalimat berikut:

Fakultas Ilmu Komputer
Universitas Indonesia

38



Motivasi untuk Sistem Tipe (Simply Typed Lambda Calculus)

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

27

Sebuah Paradox

Perhatikan kalimat berikut:

Saya seorang pembohong

38

Fakultas Ilmu Komputer
Universitas Indonesia



Russel's Paradox

Motivasi untuk Sistem Tipe (Simply Typed Lambda Calculus)

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

28

Barber Paradox

A barber (who is a man) shaves all and only those men who do not shave themselves. Does he shave himself?

Fakultas Ilmu Komputer
Universitas Indonesia

38



Russel's Paradox

Motivasi untuk Sistem Tipe (Simply Typed Lambda Calculus)

FunPro 2020

AA

- Pengantar
 - Terminologi
 - β (beta) Reduction
 - α (alpha) conversion
- Evaluasi dan Substitusi
- Representasi
 - Church Numerals
 - Penjumlahan
 - Perkalian
 - Church Boolean
 - True False
 - Operasi Logika
 - Equality dan inequalities
 - Rekursif
- Type System
 - Abstract Syntax
 - Type Judgement
 - Inference System
- Latihan
 - Beta Reduction
 - Type Systems

Barber Paradox

A barber (who is a man) shaves all and only those men who do not shave themselves. Does he shave himself?

Naïve Comprehension (NC)

$$(NC) \exists A. \forall x. (x \in A \equiv \Phi)$$

where A is not free in the formula Φ . This says, "There is a set A such that for any object x , x is an element of A if and only if the condition expressed by Φ holds." Russell's paradox arises by taking Φ to be the formula: $x \notin x$.



Abstract Syntax

Simple Type λ calculus

FunPro 2020

AA

- Pengantar
 - Terminologi
 - β (beta) Reduction
 - α (alpha) conversion
- Evaluasi dan Substitusi
- Representasi
 - Church Numerals
 - Penjumlahan
 - Perkalian
 - Church Boolean
 - True False
 - Operasi Logika
 - Equality dan inequalities
 - Rekursif
- Type System
 - Abstract Syntax
 - Type Judgement
 - Inference System
- Latihan
 - Beta Reduction
 - Type Systems

29

38

$$\begin{array}{ll} T, U & \in \text{Type} \\ T, U & ::= T \rightarrow U \\ & \dots \\ x, y, z & \in \text{Var} \\ t, u & \in \text{Term} \\ t, u & ::= x \\ & | tu \\ & | \lambda x : T . t \\ & \dots \end{array}$$

Simbol ... menyatakan bagian dari bahasa atau tipe yang relatif sudah kita kenal dan tidak membutuhkan *pure – types*.

Pada Simply Type λ Calculus, tipe dinyatakan sebagai sebuah elemen, dalam paparan kali ini dengan huruf T , dan U . Penulisan $x : T$ dibaca sebagai sebuah term atau variable x bertipe T . Tipe dapat juga berupa aplikasi fungsi diantara dua tipe $T \rightarrow U$.

Berbeda dengan asosiatif pada fungsi aplikasi, pada penulisan tipe asosiatif yang digunakan adalah asosiatif ke kanan.



Type Judgement

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Dengan adanya penerapan sistem tipe (*type system*) sebuah prosedur pemeriksaan tipe akan dibutuhkan. Prosedur tersebut di literatur disebut dengan istilah *type judgment*.

Prosedur *type judgment* memiliki tiga komponen yaitu:

- ▶ *type environment*, berisi semacam *dictionary* yang memetakan variable yang berada dalam *term* ke *type*.
- ▶ *term* yang perlu ditentukan jenis nya (*type*)
- ▶ usulan *type* untuk *term*

Type Judgment biasanya ditulis secara formal sebagai berikut:

$$\Gamma \vdash t : T$$

dengan Γ adalah *type environment*, t adalah *term* dan T adalah *type*.

Penulisan $\Gamma, x : T$ artinya, *type environment* menyatakan bahwa x bertipe T dan tipe dari variable lain masih berada dalam Γ .



Inference System

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi
 β (beta) Reduction
 α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals
Penjumlahan
Perkalian
Church Boolean
True False
Operasi Logika
Equality dan inequalities
Rekursif

Type System

Abstract Syntax
Type Judgement
Inference System

Latihan

Beta Reduction
Type Systems

Dalam *type system* untuk melakukan validitas dari tipe sesuai dengan *type judgment*, dibuatlah *inference system* yang biasanya terdiri dari:

- ▶ Sebuah *axiom* yang berisi sebuah *type judgment*, yang biasanya buat garis lurus di atasnya.
- ▶ sebuah *inference rule* terdiri dari sebuah *type judgment* (yang menyatakan kesimpulan), yang diberi garis di atasnya. Di atas garis bisa berada satu atau lebih *type judgments* yang disebut *premises*). Sebuah axiom bisa dilihat sebagai *inference rule* tanpa premises.

31

38



Inference System

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

32

Latihan

Beta Reduction

Type Systems

38

Fakultas Ilmu Komputer
Universitas Indonesia

$$\overline{\Gamma, x : T \vdash x : T} \quad (1)$$

$$\frac{\Gamma \vdash t : U \rightarrow T \quad \Gamma \vdash u : U}{\Gamma \vdash tu : T} \quad (2)$$

$$\frac{\Gamma, x : U \vdash t : T}{\Gamma \vdash (\lambda x : U. t) : U \rightarrow T} \quad (3)$$

Untuk lambda calculus yang sudah berikan ekstension literal terkait numeric aritmatika (dalam contoh ini simbol operator $+$ dan tipe *Num*) akan memiliki tambahan aturan:

$$\overline{\Gamma \vdash c : \mathbf{Num}} \quad (4)$$

$$\frac{\Gamma \vdash t : \mathbf{Num} \quad \Gamma \vdash u : \mathbf{Num}}{\Gamma \vdash t + u : \mathbf{Num}} \quad (5)$$

Sebuah penetapan tipe yang tidak bisa diturunkan menggunakan sistem tersebut biasa disebut sebagai *ill-typed*.



Contoh: type inferencing

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

33

Latihan

Beta Reduction

Type Systems

38

Fakultas Ilmu Komputer
Universitas Indonesia

$avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash (\lambda f : \text{Num} \rightarrow \text{Num} . f\ 3)(avg\ 2) : \text{Num}$

$$\frac{\frac{avg : \dots, f : \beta \vdash f : \delta \rightarrow \alpha}{avg : \dots, f : \beta \vdash f\ 3 : \alpha} \quad \frac{avg : \dots, f : \dots \vdash 3 : \delta}{avg : \dots \vdash (\lambda f : \beta \rightarrow \alpha) . f\ 3 : \beta \rightarrow \alpha}}{avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash (\lambda f : \beta \rightarrow \alpha) . f\ 3 : \beta \rightarrow \alpha} \quad \frac{avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash avg : \gamma \rightarrow \beta \quad avg : \dots \vdash 2 : \gamma}{avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash avg\ 2 : \beta}}{avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash (\lambda f : \beta \rightarrow \alpha) . f\ 3 (avg\ 2) : \alpha}$$



Contoh: type inferencing (lanj.)

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

34

Latihan

Beta Reduction

Type Systems

38

Fakultas Ilmu Komputer
Universitas Indonesia

$$\frac{\frac{\text{avg} : \dots, f : \text{Num} \rightarrow \text{Num} \vdash f : \text{Num} \rightarrow \text{Num}}{\text{avg} : \dots, f : \text{Num} \rightarrow \text{Num} \vdash f 3 : \text{Num}} \quad \text{avg} : \dots, f : \dots \vdash 3 : \text{Num}}{\text{avg} : \dots \vdash (\lambda f : \text{Num} \rightarrow \text{Num} \cdot f 3) : \text{Num} \rightarrow \text{Num}} \quad \frac{\text{avg} : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash \text{avg} : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num}}{\text{avg} : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash \text{avg} 2 : \text{Num}} \quad \text{avg} : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash (\lambda f : \text{Num} \rightarrow \text{Num} \cdot f 3)(\text{avg} 2) : \text{Num}$$

Bentuk penulisan seperti ini biasa disebut *type inference tree*.



Contoh: ill typed

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Ada kasus ketika *type inference tree* tidak sampai pada aksiom. Untuk kasus tersebut kita menyatakan bahwa type yang hendak diuji sebagai *ill typed*. Perhatikan bagian kanan tidak mencapai aksiom, ditandai dengan tidak ada garis diatas nya.

$$\frac{\frac{\frac{avg : \dots, f : \text{Num} \rightarrow \text{Num} \vdash f : \text{Num} \rightarrow \text{Num}}{avg : \dots, f : \text{Num} \rightarrow \text{Num} \vdash f 3 : \text{Num}} \quad \frac{avg : \dots, f : \dots \vdash 3 : \text{Num}}{avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash avg : \text{Num}}}{avg : \dots \vdash (\lambda f : \text{Num} \rightarrow \text{Num} \cdot f 3) : \text{Num} \rightarrow \text{Num}} \quad \frac{}{avg : \text{Num} \rightarrow \text{Num} \rightarrow \text{Num} \vdash (\lambda f : \text{Num} \rightarrow \text{Num} \cdot f 3) avg : \text{Num}}$$

35

38



Contoh: mengusul kan tipe baru

Simple Type λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

Fakultas Ilmu Komputer
Universitas Indonesia

Kadang kita belum tahu semua kemungkinan tipe, kita dapat mengusulkan perkiraan tipe terlebih dahulu kemudian menguji apakah usulan tersebut dalam membentuk susunan yang valid atau *ill typed*. Biasanya menggunakan huruf kecil Yunani. Dalam contoh ini kita akan dapati: $\alpha = \gamma = \delta = \mathbf{Num}$

$$\frac{\frac{\frac{avg : \dots, f : \beta \vdash f : \delta \rightarrow \alpha}{avg : \dots, f : \beta \vdash f 3 : \alpha}}{avg : \dots \vdash (\lambda f : \beta \cdot f 3) : \beta \rightarrow \alpha}}{avg : \mathbf{Num} \rightarrow \mathbf{Num} \rightarrow \mathbf{Num} \vdash (\lambda f : \beta \cdot f 3)(avg 2) : \alpha} \quad \frac{\frac{avg : \dots, f : \dots \vdash 3 : \delta}{avg : \mathbf{Num} \rightarrow \mathbf{Num} \rightarrow \mathbf{Num} \vdash avg : \gamma \rightarrow \beta} \quad \frac{avg : \dots \vdash 2 : \gamma}{avg : \mathbf{Num} \rightarrow \mathbf{Num} \rightarrow \mathbf{Num} \vdash avg 2 : \beta}}{avg : \mathbf{Num} \rightarrow \mathbf{Num} \rightarrow \mathbf{Num} \vdash avg 2 : \beta}$$

36

38



Latihan: β -reduction

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

1. $(\lambda x.(xy))(\lambda z.z)$

37

38

Fakultas Ilmu Komputer
Universitas Indonesia



Latihan: β -reduction

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

1. $(\lambda x.(xy))(\lambda z.z)$

2. $(\lambda x.(\lambda y.xy)x)(\lambda z.w)$

37

38



Latihan: β -reduction

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

1. $(\lambda x.(xy))(\lambda z.z)$
2. $(\lambda x.(\lambda y.xy)x)(\lambda z.w)$
3. $(\lambda f.(\lambda g.(ff)g)(\lambda h.kh))(\lambda xy.y)$

37

38



Latihan: β -reduction

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

1. $(\lambda x.(xy))(\lambda z.z)$
2. $(\lambda x.(\lambda y.xy)x)(\lambda z.w)$
3. $(\lambda f.(\lambda g.(ff)g)(\lambda h.kh))(\lambda xy.y)$
4. $((\lambda z.z)(\lambda y.yy))((\lambda x.x)a)$

37

38



Latihan: β -reduction

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

1. $(\lambda x.(xy))(\lambda z.z)$
2. $(\lambda x.(\lambda y.xy)x)(\lambda z.w)$
3. $(\lambda f.(\lambda g.(ff)g)(\lambda h.kh))(\lambda xy.y)$
4. $((\lambda z.z)(\lambda y.yy))((\lambda x.x)a)$
5. $(\lambda xy.xy)yz$

37

38



Latihan: Type System

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan

Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

38

Buktikan bahwa tipe berikut ini benar atau tidak.

1. $(\lambda(x : \text{int}).(\lambda(y : \text{int}).x + y))$

38



Latihan: Type System

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

38

Buktikan bahwa tipe berikut ini benar atau tidak.

1. $(\lambda(x : \text{int}).(\lambda(y : \text{int}).x + y))$
2. $(\lambda(f : ((\text{int} \rightarrow \text{string}) \rightarrow \text{int} \rightarrow \text{string})))(g : (\text{int} \rightarrow \text{string}))(x : \text{int}).((fg)x)$

38



Latihan: Type System

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

38

Buktikan bahwa tipe berikut ini benar atau tidak.

1. $(\lambda(x : int).(\lambda(y : int).x + y))$
2. $(\lambda(f : ((int \rightarrow string) \rightarrow int \rightarrow string)))(g : (int \rightarrow string))(x : int).((fg)x)$
3. $(\lambda(f : (int \rightarrow int \rightarrow string)))(g : (int \rightarrow int))(x : int)(y : int).((fx)(gy))$

38



Latihan: Type System

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

38

Buktikan bahwa tipe berikut ini benar atau tidak.

1. $(\lambda(x : int).(\lambda(y : int).x + y))$
2. $(\lambda(f : ((int \rightarrow string) \rightarrow int \rightarrow string)))(g : (int \rightarrow string))(x : int).((fg)x)$
3. $(\lambda(f : (int \rightarrow int \rightarrow string)))(g : (int \rightarrow int))(x : int)(y : int).((fx)(gy))$
4. $(\lambda(f : ((string \rightarrow int \rightarrow string) \rightarrow int \rightarrow string)))(x : (int \rightarrow string))(a : int)(y : (string \rightarrow int \rightarrow string)).(y(xa))$

38



Latihan: Type System

λ calculus

FunPro 2020

AA

Pengantar

Terminologi

β (beta) Reduction

α (alpha) conversion

Evaluasi dan
Substitusi

Representasi

Church Numerals

Penjumlahan

Perkalian

Church Boolean

True False

Operasi Logika

Equality dan inequalities

Rekursif

Type System

Abstract Syntax

Type Judgement

Inference System

Latihan

Beta Reduction

Type Systems

38

Buktikan bahwa tipe berikut ini benar atau tidak.

1. $(\lambda(x : int).(\lambda(y : int).x + y))$
2. $(\lambda(f : ((int \rightarrow string) \rightarrow int \rightarrow string)))(g : (int \rightarrow string))(x : int).((fg)x)$
3. $(\lambda(f : (int \rightarrow int \rightarrow string)))(g : (int \rightarrow int))(x : int)(y : int).((fx)(gy))$
4. $(\lambda(f : ((string \rightarrow int \rightarrow string) \rightarrow int \rightarrow string))(x : (int \rightarrow string))(a : int)(y : (string \rightarrow int \rightarrow string)).(y(xa)))$
5. $(\lambda(x : (int \rightarrow int))(y : int).xx(y + 3))$

38