MADC 101R04/MCSE 101R04
Logic for Computer Science Engineers
Unit IV
λ Calculus

Introduction

- What is Computation?

Introduction

- What is Computation?
- What machines can do, and what not
- ▸ Link
- What is Lambda Calculus?

Introduction

- What is Computation?
- What machines can do, and what not
- ▸ Link
- What is Lambda Calculus?
- Lambda Calculus does exactly the same thing, but without wheels
- Mathematical from a distance-Just only a Greek letter in it
- Easy to understand
- Much better intuition of computation

Short History...

- Alonzo Church - 1930's
- Calculus - Don't be alarmed - does not have any complicated formulae or operations
- Take a line of letters (or symbols), and performing a little cut and paste operation on it
- Lambda Calculus can compute everything that can be computed, just with a very simple cut and paste

Introduction

- λ calculus: A collection of several formal systems based on notation invented by Alonzo Church

Introduction

- λ calculus: A collection of several formal systems based on notation invented by Alonzo Church
- Why λ?

Introduction

- λ calculus: A collection of several formal systems based on notation invented by Alonzo Church
- Why λ?
- Accident

Introduction

- λ calculus: A collection of several formal systems based on notation invented by Alonzo Church
- Why λ?
- Accident
- Initially $\hat{y}$, wrongly typed as ^y and the type setter typed as λ

Introduction

- λ calculus: A collection of several formal systems based on notation invented by Alonzo Church
- Why λ?
- Accident
- Initially $\hat{y}$, wrongly typed as ^y and the type setter typed as λ
- Describe the most basic ways that operators or functions can be combined to form other operators

Note

- Each λ system has a different structure depending on the use
- May have extra constant symbols
- Most have syntactic restrictions-type restrictions
- Simple syntax

Church's Notation

- A systematic way of constructing for each expression involving "x" a notation for the corresponding function of x
- Let us consider two functions f:x$\rightarrow$x-y and g:y$\rightarrow$x-y. It can be written as f=λx.x-y and g=λy.x-y

Church's Notation

- A systematic way of constructing for each expression involving "x" a notation for the corresponding function of x
- Let us consider two functions f:x$\rightarrow$x-y and g:y$\rightarrow$x-y. It can be written as f=λx.x-y and g=λy.x-y
- A function starts always with the λ and a variable, followed by a dot, and then comes an expression
- The λ does not have any complicated meaning: it just says that a function starts here
- The λ-variable-. part of a function is called its head, and the remainder (the expression) is called the body
- Example:
  - From the previous slide, f(0)=0-y and f(1)=1-y
  - In the λ notation, the above becomes, (λx.x-y)(0)=0-y and (λx.x-y)(1)=1-y

A Note about some key terms...

- What is the value or meaning of a variable?

A Note about some key terms...

- What is the value or meaning of a variable?
- Nothing. They are just empty names. Even the name is unimportant. The only thing is: when two variables have the same name, they are the same. You can rename variables all you want, without changing the expression

A Note about some key terms...

- What is the value or meaning of a variable?
- Nothing. They are just empty names. Even the name is unimportant. The only thing is: when two variables have the same name, they are the same. You can rename variables all you want, without changing the expression
- Clumsier than others
- Intended for denoting higher order functions not just function of no.s
- Notation is systematic and hence suitable for incorporating into a programming language

Functions of more than one variable

- Let us consider the functions h(x,y)=x-y and k(y,x)=x-y
- The above can be denoted as h=λxy.x-y and k=λyx.x-y
- The above can be combined to form

Functions of more than one variable

- Let us consider the functions h(x,y)=x-y and k(y,x)=x-y
- The above can be denoted as h=λxy.x-y and k=λyx.x-y
- The above can be combined to form
- h*=λx(λy.x-y)

Functions of more than one variable

- Let us consider the functions h(x,y)=x-y and k(y,x)=x-y
- The above can be denoted as h=λxy.x-y and k=λyx.x-y
- The above can be combined to form
- h*=λx(λy.x-y)
- For each number 'a' we have h*(a)=λy.a-y

Functions of more than one variable

- Let us consider the functions h(x,y)=x-y and k(y,x)=x-y
- The above can be denoted as h=λxy.x-y and k=λyx.x-y
- The above can be combined to form
- h*=λx(λy.x-y)
- For each number 'a' we have h*(a)=λy.a-y
- Hence for each pair of numbers a,b,(h*(a))(b) = (λy.a-y)(b)=a-b=h(a,b)

Note

- h* can be viewed as representing 'h'
- function means function in one variable
- Use of h* instead of h $\rightarrow$ currying

Note

- M is a function or operator $\rightarrow$ (MN) is the result of applying M to argument N if exists
- (λx.M)$\rightarrow$Operator or function whose value at an argument N is calculated by substituting N for x in M

Note

- M is a function or operator $\rightarrow$ (MN) is the result of applying M to argument N if exists
- (λx.M)$\rightarrow$Operator or function whose value at an argument N is calculated by substituting N for x in M
  - (λx.x(xy))N =N(Ny)

Note

- M is a function or operator $\rightarrow$ (MN) is the result of applying M to argument N if exists
- $(\lambda x.M) \rightarrow$ Operator or function whose value at an argument N is calculated by substituting N for x in M
  - $(\lambda x.x(xy))N = N(Ny)$
  - $(\lambda x.y)N = y$

Term Structure and Substitution

Definition

The length of a term M called lgh(M)is the total no. of occurrences of atoms in M.

- lgh(a)=1 for atoms a
- lgh(MN)=lgh(M)+lgh(N)
- lgh(λx,M)=1+lgh(M)

- Note:
  - Induction on M means induction on lgh(M)

Term Structure and Substitution

Definition

The length of a term M called lgh(M)is the total no. of occurrences of atoms in M.

- lgh(a)=1 for atoms a
- lgh(MN)=lgh(M)+lgh(N)
- lgh(λx,M)=1+lgh(M)

- Note:
  - Induction on M means induction on lgh(M)
- Example:If M $\equiv$x(λy.yux)then lgh(M)=5

Relation "occurs in"

Definition

For λ terms P and Q the relation P "occurs in" Q(or P is a sub term of Q or Q contains P) is defined by induction on Q. Thus

- P occurs in P
- If P occurs in M or in N then P occurs in (MN)
- If P occurs in M or P$\equiv$x then P occurs in (λx.M)

Relation "occurs in"

Definition

For λ terms P and Q the relation P "occurs in" Q(or P is a sub term of Q or Q contains P) is defined by induction on Q. Thus

- P occurs in P
- If P occurs in M or in N then P occurs in (MN)
- If P occurs in M or P$\equiv$x then P occurs in (λx.M)

- Example:Let us consider the expression((xy)(λx.(xy))). Two occurrences of xy and three occurrences of x

Exercise

1. Mark all the occurrences of xy in the term λxy.xy

Exercise

1. Mark all the occurrences of xy in the term λxy.xy
   Solution: λxy.xy=(λx.(λy.($\underline{xy}$)))

Exercise

1. Mark all the occurrences of xy in the term λxy.xy
   Solution: λxy.xy=(λx.(λy.($\underline{xy}$)))
2. All occurrences of uv in x(uv)(λu.v(uv))uv

Exercise

1. Mark all the occurrences of xy in the term λxy.xy
   Solution: λxy.xy=(λx.(λy.($\underline{xy}$)))
2. All occurrences of uv in x(uv)(λu.v(uv))uv
   Solution: x(uv)(λu.v(uv))uv = ((x($\underline{uv}$))(λu.(v($\underline{uv}$))))u)v)

Exercise

1. Mark all the occurrences of xy in the term λxy.xy
   Solution: λxy.xy=(λx.(λy.($\underline{xy}$)))
2. All occurrences of uv in x(uv)(λu.v(uv))uv
   Solution: x(uv)(λu.v(uv))uv = ((x($\underline{uv}$))(λu.(v($\underline{uv}$))))u)v)
3. Does λu.u occur in λu.uv?

Exercise

1. Mark all the occurrences of xy in the term λxy.xy
   Solution: $\lambda xy.xy = (\lambda x.(\lambda y.(\underline{xy})))$
2. All occurrences of uv in x(uv)(λu.v(uv))uv
   Solution: $x(uv)(\lambda u.v(uv))uv = ((x(\underline{uv}))(\lambda u.(v(\underline{uv}))))u)v)$
3. Does λu.u occur in λu.uv?
   Solution: $\lambda u.u = (\lambda u.u)$
   $\lambda u.uv = (\lambda u.(uv))$

Exercise

1. Mark all the occurrences of xy in the term λxy.xy
   Solution: λxy.xy=(λx.(λy.($\underline{xy}$)))
2. All occurrences of uv in x(uv)(λu.v(uv))uv
   Solution: x(uv)(λu.v(uv))uv = ((x($\underline{uv}$))(λu.(v($\underline{uv}$))))u)v)
3. Does λu.u occur in λu.uv?
   Solution: λu.u = (λu.u)
   λu.uv=(λu.(uv))
   No it does not occur

Scope of a Variable

Definition

For a particular occurrence of λx.M in a term P, the occurrence of M is called the scope of occurrence of λx on the left

- Example: P$\equiv$(λx.yx(λx.y(λy.z)x))vw

Scope of a Variable

Definition

For a particular occurrence of λx.M in a term P, the occurrence of M is called the scope of occurrence of λx on the left

- Example: P$\equiv$(λx.yx(λx.y(λy.z)x))vw
  Solution: Scope of leftmost λy is yx(λx.y(λy.z)x)

Scope of a Variable

Definition

For a particular occurrence of λx.M in a term P, the occurrence of M is called the scope of occurrence of λx on the left

- Example: P$\equiv$(λx.yx(λx.y(λy.z)x))vw
  Solution: Scope of leftmost λy is yx(λx.y(λy.z)x)
  Scope of leftmost λx is y(λy.z)x

Scope of a Variable

Definition

For a particular occurrence of λx.M in a term P, the occurrence of M is called the scope of occurrence of λx on the left

- Example: P$\equiv$(λx.yx(λx.y(λy.z)x))vw
  Solution: Scope of leftmost λy is yx(λx.y(λy.z)x)
  Scope of leftmost λx is y(λy.z)x
  Scope of rightmost λy is z

Free and Bound Variables

Definition

An occurrence of a variable x in a term P is called

- Bound if it is in the scope of a λx in P
- Bond and Binding iff it is the x in λx
- Free otherwise

- If x has atleast on binding occurrence in P we call x a bound variable of P
- If x has atleast one free occurrence in P, we call x a free variable of P
- The set of all free variables of P is called FV(P)
- A closed term is a term with out any free variables

Example

- Let us consider the expression xv(λyz.yv)w

Example

- Let us consider the expression $xv(\lambda yz.yv)w$
- $((xv(\lambda y.(\lambda z.(yv))))w)$

Example

- Let us consider the expression $xv(\lambda yz.yv)w$
- $((xv(\lambda y.(\lambda z.(yv))))w)$
- free both bound and bind bound

Note

- x is both a free and bound variable of P
- This is not normally advisable in practice
- Inorder to keep λ term simple this is used

Substitution

Definition

For any M,N,x define [N/x]M to be the result of substituting N for every free occurrence of x in M and changing bound variables to avoid clashes. It is defined inductively on M as follows:

N/x $x \equiv N$

N/x $a \equiv a \ \forall$ atoms $a \neq x$

N/x $PQ \equiv ([N/x]P[N/x]Q)$

N/x $(\lambda x.P) \equiv \lambda x.P$

N/x $(\lambda .P) \equiv \lambda y.P$ if x not $\in FV(P)$

N/x $(\lambda y.P) \equiv \lambda y.[N/x]P$ if $x \in FV(P)$ and y not $\in FV(N)$

N/x $(\lambda y.P) \equiv \lambda z.[N/x][z/y]P$ if $x \in FV(P)$ and $y \in FV(N)$

Note

- For example, y not $\equiv$x in (g) z is chosen to be first variable not $\in$ FV(NP)
- (g) is to prevent the intutive meaning of $[N/x](\lambda y.P)$ from depending on the bound variable y
- Let us assume that there are three variables w,x,y such that $[w/x](\lambda y.x) \rightarrow$ constant function always x
- By (f) and (a) $[w/x](\lambda y.x) \equiv \lambda y.w$
- Similarly $[w/x](\lambda w.x)$ value is always w
- If $[w/x](\lambda w.x)$ was evaluated by (f) we would have $[w/x](\lambda w.x) = \lambda w.w$ which represents identity function and not a constant function
- By (g) $N \equiv y \equiv w$, we have $[w/x](\lambda w.x) \equiv \lambda z.[w/x][z/w]x \equiv \lambda z.[w/x]x \equiv \lambda z.w$

Exercises

- Evaluate the following expressions;

$(uv)/x \ (\lambda y.x(\lambda w.vwx))$

Exercises

- Evaluate the following expressions;

$(uv)/x \quad (\lambda y.x(\lambda w.vwx))$

Assignment

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

Exercises

- Evaluate the following expressions;

$(uv)/x$ $(\lambda y.x(\lambda w.vwx))$

Assignment

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

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

$(\lambda y.vy)/x$ $(y(\lambda vxv))$

Exercises

- Evaluate the following expressions;

$(uv)/x$ $(\lambda y.x(\lambda w.vwx))$
Assignment
$(\lambda y.xy)/x$ $(\lambda y.x(\lambda x.x))$
Solution: $\lambda y.(\lambda y.xy)(\lambda x.x)$
$(\lambda y.vy)/x$ $(y(\lambda vxv))$
Solution:$y(\lambda z.(\lambda y.vy)z)$ if $z \neq v,y,x$
$(uv)/x$ $(\lambda x.zy)$

Exercises

- Evaluate the following expressions;

$(uv)/x$ $(\lambda y.x(\lambda w.vwx))$
Assignment

$(\lambda y.xy)/x$ $(\lambda y.x(\lambda x.x))$
Solution: $\lambda y.(\lambda y.xy)(\lambda x.x)$

$(\lambda y.vy)/x$ $(y(\lambda vxv))$
Solution:$y(\lambda z.(\lambda y.vy)z)$ if $z \neq v,y,x$

$(uv)/x$ $(\lambda x.zy)$
Assignment

Lemma

For all terms M,N and variables x:

x/x $M \equiv M$

1. x *not* $\in FV(M) \implies [N/x]M \equiv M$
2. $x \in FV(M) \implies FV([N/x]M) = FV(N) \cup (FV(M) - \{x\})$
3. $lgh([g/x]M) = lgh(M)$

Lemma

Let x,y,v be distinct and let no variable bound in M be free in vPQ. Then

P/v $[v/x]M \equiv [P/x]M$ if $v \notin FV(M)$

x/v $[v/x]M \equiv M$ if $v \notin FV(M)$

P/x $[Q/y]M \equiv [([P/x]Q)/y][P/x]M$ if $y \notin FV(P)$

P/x $[Q/y]M \equiv [Q/y][P/x]M$ if $y \notin FV(P)$, $x \notin FV(Q)$

P/x $[Q/x]M \equiv [([P/x]Q/x]M$

Change of Bound Variable and Congruence

Definition

Let a term P contain an occurrence of λx.M and let y$\notin$FV(M). The act of replacing that λx.M by λy.[y/x]M is called a change of bound variable or an α conversion in P
If P can be changed to Q by a finite series of changes of bound variables, we shall say P is congruent to Q or P α converts to Q or P$\equiv_\alpha$Q

- Example: Please refer BB notes

Lemma

1. *If $P \equiv_\alpha Q$ then $FV(P)=FV(Q)$*
2. *The relation $\equiv_\alpha$ is an equivalence relation*

Lemma

$M \equiv_\alpha M'$, $N \equiv_\alpha N' \Longrightarrow [N/x]M \equiv_\alpha [N'/x]M$

Answers for the I Monthly Test Questions

1. (a)Yes it is a λ term.(0.5 Mark) No it is not encouraged(0.5 Mark)Reason: Two occurrences of λy in one term (1 Mark)
   (b)((((λa.(λb.(ac)(bc)))p)q)r) (2 Marks)
2. (a) yes(0.5 Marks) Reason: λyx.x $\equiv$ (λy.(λx.x)). Hence λx.x is contained in λyx.x (1.5 Marks)
   (b)Free Variables={a,b,q,c} Bound={right most p} Bound and Bind={leftmost p} (1+0.5+0.5=2 Marks)
3. (a) a(λz.[(λa.va)/b][z/v]bv)$\equiv$ λz.(λa.va)z (2 Marks)
   (b)(λa.(λb.ba)c)d $\equiv$(λa.[c/b]ba)d $\triangleright_{1\beta}$(λa.ca)d $\equiv$ [d/a]ca $\triangleright_{\beta}$ cd (2 Marks)

Answers for the I Monthly Test Questions

4. (a)Proving P and Q not β nf (1 Mark)
   PQ=λy.y(λuvw.w)Ω(λz.z)$\Omega \triangleright_{1\beta}$ λy.y(λuvw.w)$\Omega\Omega \triangleright_{1\beta}$ λy.y(λvw.w)$\Omega \triangleright_{\beta}$ λy.y(λw.w) which is a β nf(1 Mark)
   (b)If P, Q $\in \beta$-nf $\implies$ P and Q have no β redexes
   Also P $=_\beta$ Q. Since P and Q has no β redexes P $\equiv_\alpha$ Q (2 Marks)
5. (a) B: (B(f,g))(x)=f(g(x))(0.5 Mark)
   B':(B'(f,g))(x)=g(f(x))(0.5 Mark) K: (K(a))(x)=a (1 Mark)
   (b)All occurrences are free variables.(1 Mark) Reason: There is no λ to bind them. (1 Mark)

Answers for the I Monthly Test Questions

6. Statement (2 Marks) Proof (8 Marks)
7. (a)λab.ab$\equiv$ λa.(λb.ab)$\equiv_\alpha$ λa.(λv.av)$\equiv_\alpha$ λu.(λv.uv)$\equiv$ λuv.uv (5 Marks)
   (b) Given x $\in$ FV(M). Let FV(M)=$\{$x,...$\}$. FV([N/x]M) means replacing every free occurrence of M by N. Therefore FV(M)=FV(M)-$\{$x$\}$. Hence FV([N/x]M)=FV(N)$\cup$(FV(M)-$\{$x$\}$) (5 Marks)
8. (a) length=5 (2 Marks) (b)Yes(1 Mark) Reason (2 Marks) FV=$\{$x,a,b$\}$(1 Mark) (c) (a) S(A,I)(x)=A(x,I(x))=A(x,x)=2x (2 Marks) (b) KISKKI(2 Marks)

Weak Reduction

Definition

Any term Ix,Kxy or Sxyz is called a weak redex. Contracting an occurrence of a weak redex in a form U means replacing one occurrence of Ix by x or Kxy by x or Sxyz by xz(yz) iff this changes U to U'we say that U weakly contracts to U'or U$\triangleright_{1w}$ U'
V is obtained from U by a finite(empty) series of weak contractions, we say that U weakly reduce to V or U $\triangleright_w$ V

Definition

A term that contains no weak redexes is called a weak normal form

Examples

1. Define B$\equiv$S(KS)K. Prove that Bxyz $\triangleright_w$x(yz)
   Solution: Refer BB
2. If C$\equiv$S(BBS)(KK). Prove that Cxyz $\triangleright_w$ xzy
3. SIKx (Assignment)
4. SSKxy (Assignment)
5. S(SK)xy (Assignment)
6. S(KS)Sxyz (Assignment)

Abstraction in CL

- Define a CL-term called '[x].M' for every x and M, with the property that $([x].M)N \triangleright_w [N/x]M$
- The term [x].M will play a role of $(\lambda x.M)$
- It will be a combination of I's, K's, S's and parts of M

Abstraction

Definition

For every CL-term M and every variable x, a CL-term called [x].M is defined by induction on M, thus:

1. $[x].M \equiv KM$ if $x \notin FV(M)$
2. $[x].x \equiv I$
3. $[x].Ux \equiv U$ if $x \notin FV(U)$
4. $[x].UV \equiv S([x].U)([x].V)$ if neither (a) nor (c) applies

Example

- Prove that $[x].xy \equiv SI(Ky)$
- Solution: Refer BB
- Note:
  1. In λ-calculus an expression λx can be part of a λ-term, for example the term $\lambda x.xy$
  2. But in CL, the corresponding expression $[x]$ is not part of the formal language of CL-terms at all
  3. In the previous example, the expression $[x].xy$ is not itself a CL-term, but is merely a short-hand to denote the CL-term $SI(Ky)$

Theorem

Theorem

The clauses in the definition of abstraction allow us to construct $[x].M$ for all x and M. Further, $[x].M$ does not contain x, and, for all N, $([x].M)N \triangleright_w [N/x]M$

Lemma (Substitution lemma for $\triangleright_w$)

(a)$X \triangleright_w Y \implies FV(X) \supseteq FV(Y)$
(b) $X \triangleright_w Y \implies [X/v]Z \triangleright_w [Y/v]Z$
(c) $X \triangleright_w Y \implies [U1/x1, \ldots, Un/xn]X \triangleright_w [U1/x1, \ldots, Un/xn]Y$

Theorem

Proof

By induction on M we shall prove that $[x].M$ is always defined, does not contain x, and that $([x].M)\ x \triangleright_w M$

The theorem will follow by substituting N for x and using (c) of the previous lemma.

Case 1: $M \equiv x$. Then definition of abstraction(b) applies, and $([x].x)\ x \equiv I\ x \triangleright_w x$

Case 2: M is an atom and $M \not\equiv x$. Then definition of abstraction (a) applies, and $([x].M)\ x \equiv KMx \triangleright_w M$

Case 3: $M \equiv UV$. By the induction hypothesis, we may assume $([x].U)\ x \triangleright_w U$, $([x].V\ )\ x \triangleright_w V$

Subcase 3(i): $x \notin FV(M)$. Like Case 2.

Theorem Contd..

Proof.

Subcase 3(ii): $x \notin FV(U)$ and $V \equiv x$. Then
$([x].M)\ x \equiv ([x].Ux)\ x$
$\equiv Ux$ by(c)
$\equiv M$
Subcase 3(iii): Neither of the above two subcases applies. Then
$([x].M)\ x \equiv S([x].U)([x].V)\ x$ by definition(f)
$\triangleright_{1w}$ $([x].U)\ x\ (([x].V)\ x)$
$\triangleright_{1w}$ UV (by induction hypothesis)
$\equiv M$ $\square$

Problems

1. Evaluate: [x].u(vx)
   Solution: Refer BB
2. Evaluate: [x].x(Sy)
   Solution: Refer BB
3. Evaluate: [x].uxxv
   Assignment

Definition

For all variables $x_1,\ldots,x_n$ (not necessarily distinct), $[x_1,\ldots,x_n].M \equiv [x_1].([x_2].(\ldots([x_n].M)\ldots))$.

Problems

1. $[x, y]$.x Refer BB
2. $[x, y, z]$.xz(yz) Refer BB
3. $[x, y, z]$.xzy Assignment
4. $[x, y, z]$.y(xz) Assignment
5. $[x, y]$.xyy Assignment

Weak Equality

- We shall say X is weakly equal or weakly convertible to Y , or X $=_w$ Y , iff Y can be obtained from X by a finite (perhaps empty) series of weak contractions and reversed weak contractions. That is, X $=_w$ Y iff there exist X_0 , ... ,X_n (n$\geq$ 0) such that ($\forall$i $\geq$ n 1) ($X_i \triangleright_{1w} X_{i+1}$ or $X_{i+1} \triangleright_{1w} X_i$), $X_0 \equiv$ X, $X_n \equiv$ Y.

Problem

- Prove that BWBIx$=_w$SIIx
- Solution Refer BB

Power of λ and Combinators

- Express power of both λ and CL
- Fixed-point theorem, Bohms theorem, and a theorem which helps in proving that a term has no normal form
- Neutral Notations are used

Notation	Meaning for λ	Meaning for CL
term	λterm	CL Term
$X \equiv Y$	$X \equiv_{\alpha} Y$	X is identical to Y
$X \triangleright_{\beta,w} Y$	$X \triangleright_{\beta} Y$	$X \triangleright_{w} Y$
$X =_{\beta,w} Y$	$X =_{\beta} Y$	$X =_{w} Y$
λx	λx	[x]

Combinator

Definition

A combinator is (in λ) a closed pure term, i.e. a term containing neither free variables nor atomic constants, and (in CL) a term whose only atoms are the basic combinators I, K, S. In λ, the following combinators are given special names:
B $\equiv$ λxyz.x(yz), B' $\equiv$ λxyz.y(xz), C$\equiv$ λxyz.xzy, I $\equiv$ λx.x, K $\equiv$ λxy.x, S $\equiv$ λxyz.xz(yz), W $\equiv$ λxy.xyy

Fixed Point Theorem

Definition

A fixed point of an operator or function is an object which does not change when the operator is applied to it. For example, the operation of squaring numbers has two fixed points 0 and 1, since $0^2 = 0$ and $1^2 = 1$; and the successor-function has none, since $n + 1 \neq n$ for all n.

Fixed Point Theorem

Theorem

In both λ and CL, there is a combinator Y such that (a)$Yx =_{\beta,w} x(Yx)$. In fact, there is a Y with the stronger property (b)$Yx \triangleright_{\beta,w} x(Yx)$.

Proof.

Proof Refer BB $\square$

Corollary

Corollary

In λ and CL: for every Z and $n \geq 0$, the equation $xy_1 \ldots y_n = Z$ can be solved for x. That is, there is a term X such that $Xy_1 \ldots y_n =_{\beta,w} [X/x]Z$

Corollary (Double Fixed Point Theorem)

In λ and CL: for every pair of terms X, Y there exist P , Q such that $XP\ Q =_{\beta,w} P$, $Y\ P\ Q =_{\beta,w} Q$.

Fixed Point Combinator

Definition

A fixed-point combinator is any combinator Y such that YX $=_{\beta,w}$ X(YX) for all terms X. Define $Y_{Turing} \equiv$ U U , where U $\equiv$ λux.x(uux), $Y_{CurryRos} \equiv \lambda$x.V V , where V $\equiv \lambda$y.x(yy).

Problems

1. Prove that $Y_{Curry Ros}$ is a fixed-point combinator
2. Prove that the following terms are fixed-point combinators (in both λ and CL): $Y_0 \equiv WS(BWB)$, $Y_1 \equiv WI(B(SI)(WI))$(Assignment)
3. Prove that if a term Y is a fixed-point combinator, then (i) SIY $=_{\beta,w}$ Y (and so SIY is a fixed-point combinator), (ii) Y (SI) is a fixed-point combinator.(Assignment)

Bohms theorem

- The members of a significant class of normal forms can be distinguished from each other in a very powerful way
- It is due to Corrado Bohm, and has applications in both the syntax and semantics of λ and CL

β η normal forms

Definition

In λ-calculus, a term of form λx.M x with x$\notin$ FV(M) is called an η-redex and is said to η-contract to M. A λ-term X which contains no β-redexes and no η-redexes is called a $\beta\eta$-normal form. The class of all such λ-terms is called $\beta\eta$-nf or $\lambda\beta\eta$-nf

Strong Normal Form

Definition

In CL, the class strong nf is defined inductively as follows. Its members are called strong normal forms
(a) All atoms other than I, K and S are in strong nf;
(b) If $X_1, \ldots, X_n$ are in strong nf, and a is any atom $\equiv$ I, K, S, then $aX_1 \ldots X_n$ is in strong nf
(c) If X is in strong nf, then so is [x].X

Bohm's Theorem

Theorem

In λ and CL: let M and N be combinators, either in $\beta\eta$-normal form (in λ) or in strong normal form (in CL). If $M \not\equiv N$, then there exist $n \geq 0$ and combinators $L_1 , \ldots , L_n$ such that $ML_1 \ldots L_n xy \triangleright_{\beta,\omega} x$, $N L_1 \ldots L_n xy \triangleright_{\beta,\omega} y$.

Representing Computable functions

Definition (Church Numerals)

For every n $\in \mathbb{N}$, the Church numeral for n is a term we shall call $\bar{n}$ defined (in) by (a)$\bar{n} \equiv \lambda$xy.x^n y, and (in CL) by (b)$\bar{n} \equiv$(SB)^n (KI) (B $\equiv$ S(KS)K).

- Note: In both λ and CL, the Church numerals have the useful property that, for all terms F, X, $\bar{n}$ F X $\triangleright_{\beta,w}$ F^n X.

Representability

Definition (Representability)

Let ϕ be an n-argument partial function, i.e. a function from a subset of $\mathbb{N}^n$ into $\mathbb{N}$ (n ≥ 0). A term X in λ or CL is said to represent ϕ iff, for all $m_1, \ldots, m_n \in \mathbb{N}$, (a) $\phi(m_1, \ldots, m_n) = p \implies X\ \bar{m}_1 \ldots \bar{m}_n =_{\beta,w} \bar{p}$, and (b) $\phi(m_1, \ldots, m_n)$ does not exist $\implies X\bar{m}_1 \ldots \bar{m}_n$ has no nf.