Hoofdstuk 3: Lambdarekenen:

Oefening 1:

(a) xyz

(b) x(yz)

(c) λx.xy

(d) (λx.x)y

(e) λxy.z

(f) λxy.yx

(g) λx.(λy.y)x

(h) (λxy.y)x

(i) λx.(λy.xy)x

(j) λx.y(λxy.xz(λz.xyz))yz

Oefening 2:

(a) LMN

(b) L(MN)

(c) MNPQ

(d) M(NPQ)

Oefening 3:

(a) {x, y, z}

(b) {x, y, z}

(c) {y}

(d) {y}

(e) {z}

(f) /

(g) /

(h) {x}

(i) /

(j) {y, z}

Oefening 4:

(a) (λx.x)y

(b) ((λx.x)y)

(c) ((λx.x)y)

(d) (λx.xy)

(e) (λx.x(λx.xy))

(f) (λx.xy)

Oefening 5:

(a) (λx.x)yz = <β-conversie> x [x := yz]
= <Svar> yz

(b) (λx.xy)(λz.z) = <β-conversie> (xy) [x := λz.z]
= <Sapp, Svar> (λz.z)y
= <β-conversie> z [z := y]
= <Svar> y

(c) (λxy.x)(λz.z) = <haakjes> (λx.(λy.x))(λz.z)
= <β-conversie> (λy.x) [x := λz.z]
= <Sabs, Svar> λp.(λz.z)

(d) (λxyz.xz(yz))(λxy.x)(λxy.x) = <haakjes> ((λx.(λy.(λz.xz(yz))))(λx.(λy.x)))(λx.(λy.x))
= <β-conversie> (λy.(λz.xz(yz))) [x := λx.(λy.x)] (λx.(λy.x))
= <Sabs> (λy.(λz.((xz)(yz))[x := λx.(λy.x)])) (λx.(λy.x))
= <Sapp> (λy.(λz.((xz)[x := λx.(λy.x)](yz) [x := λx.(λy.x)]))
(λx.(λy.x))
= <Sapp, Svar> (λy.(λz.(((λx.(λy.x))z)(yz)))) (λx.(λy.x))
= <β-conversie> (λy.(λz.((λy.x) [x := z] (yz)))) (λx.(λy.x))
= <Sabs, Svar> (λy.(λz.((λy.z) (yz)))) (λx.(λy.x))
= <β-conversie> (λy.(λz.(z [y := yz]))) (λx.(λy.x))
= <Svar> (λy.(λz.(z))) (λx.(λy.x))
= <β-conversie> (λz.z) [y := (λx.(λy.x))]
= <Sabs, Svar> λz.z

(e) $(\lambda x.((\lambda y.(yx))(\lambda z.z)))p$ = <β-conversie> $((\lambda y.(yx))(\lambda z.z))\ [x := p]$
= <Sapp> $((\lambda y.(yx))\ [x := p]\ (\lambda z.z)\ [x := p]$
= <Sabs> $((\lambda y.((yx)\ [x := p])))\ (\lambda z.(z\ [x := p]))$
= <Sapp> $((\lambda y.(y\ [x := p]\ x\ [x := p]\ )))\ (\lambda z.(z\ [x := p]))$
= <Svar> $((\lambda y.(yp)))\ (\lambda z.(z))$
= <haakjes> $(\lambda y.(yp))\ (\lambda z.z)$
= <β-conversie> $(yp)\ [y := \lambda z.z]$
= <Saps> $(y\ [y := \lambda z.z]\ p\ [y := \lambda z.z])$
= <Svar> $(\lambda z.z)\ p$
= <β-conversie> $z\ [z := p]$
= <Svar> p

(f) $(\lambda z.(z(zx)))(\lambda y.(yp))$ = <β-conversie> $(z(zx))\ [z := \lambda y.(yp)]$
= <Sapp> $(z\ [z := \lambda y.(yp)])\ (zx)\ [z := \lambda y.(yp)]$
= <Sapp> $(z\ [z := \lambda y.(yp)])\ ((z\ [z := \lambda y.(yp)])\ (x\ [z := \lambda y.(yp)]))$
= <Svar> $(\lambda y.(yp))\ ((\lambda y.(yp))x)$
= <β-conversie> $(yp)\ [y := (\lambda y.(yp))x]$
= <Sapp> $(y\ [y := (\lambda y.(yp))x]\ p\ [y := (\lambda y.(yp))x])$
= <Svar> $(((\lambda y.(yp))x)\ p)$
= <β-conversie> $(yp)\ [y := x]\ p$
= <Sapp> $(y\ [y := x]\ p\ [y := x])\ p$
= <Svar> xpp

(g) $(\lambda x.(\lambda xy.yx)xx)(\lambda zx.z(zx))$ = <haakjes> $(\lambda x.((\lambda x.(\lambda y.yx))x)x)(\lambda z.(\lambda x.z(zx)))$
= <β-conversie> $(((\lambda x.(\lambda y.yx))x)x)\ [x := \lambda z.(\lambda x.z(zx))]$
= <Sabs> $((\lambda x.(\lambda y.yx))xx)$
= <β-conversie> $(\lambda y.yx)\ [x := xx]$
= <Sabs> $(\lambda y.((yx)\ [x := xx]))$
= <Sapp> $(\lambda y.((y\ [x := xx])\ (x\ [x := xx])))$
= <Svar> $\lambda y.((y)(xx))$

<u>Oefening 6:</u>

$(\lambda yz.zy)((\lambda x.xxx)(\lambda x.xxx))(\lambda xy.y)$ = <haakjes> $(\lambda y.(\lambda z.zy))((\lambda x.xxx)(\lambda x.xxx))(\lambda x.(\lambda y.y))$
= <β-conversie> $(\lambda z.zy)\ [y := (\lambda x.xxx)(\lambda x.xxx)]\ (\lambda x.(\lambda y.y))$
= <Sabs> $(\lambda z.((zy)\ [y := (\lambda x.xxx)(\lambda x.xxx)])\ (\lambda x.(\lambda y.y))$
= <Sapp> $(\lambda z.((z\ [y := (\lambda x.xxx)(\lambda x.xxx)])(y\ [y :=$
$(\lambda x.xxx)(\lambda x.xxx)]))\ (\lambda x.(\lambda y.y))$
= <Svar> $(\lambda z.((z)((\lambda x.xxx)(\lambda x.xxx)))\ (\lambda x.(\lambda y.y))$
= <β-conversie> $((z)((\lambda x.xxx)(\lambda x.xxx)))\ [z := \lambda x.(\lambda y.y)]$
= <Sapp> $((z)\ [z := \lambda x.(\lambda y.y)]\ ((\lambda x.xxx)(\lambda x.xxx)))$
$[z := \lambda x.(\lambda y.y)])$
= <Svar> $((\lambda x.(\lambda y.y))((\lambda x.xxx)(\lambda x.xxx)))$
= <β-conversie> $(\lambda y.y)\ [x := (\lambda x.xxx)(\lambda x.xxx)]$
= <Sabs> $(\lambda y.((y)\ [x := (\lambda x.xxx)(\lambda x.xxx)])$
= <Svar> $\lambda y.y$

Oefening 7:

(a) A p3 p2 ≡ <def> λxypq.xp(ypq) p3 p2
≡ <β-conversie, def p3> (λypq.((λxy.x(x(xy)))p)(ypq)) p2
≡ <β-conversie> (λypq.((λy.p(p(py))))(ypq)) p2
≡ <β-conversie> (λypq.(p(p(p(ypq))))) p2
≡ <β-conversie, def p2> (λpq.(p(p(p((λxy.x(xy))pq)))))
≡ <β-conversie> (λpq.(p(p(p((λy.p(py))q)))))
≡ <β-conversie> (λpq.(p(p(p(p(pq))))))
≡ <α-conversie> (λxy.(x(x(x(x(xy))))))

(b) P p3 p2 ≡ <def> (λxyz.x(yz)) p3 p2
≡ <β-conversie, def p3> (λyz.((λxy.x(x(xy)))(yz))) p2
≡ <β-conversie> (λyz.(λy.(yz)((yz)((yz)y)))) p2

??????????????

(c) E p3 p2 ≡

Oefening 8:

- Head(Cons P Empty) =
- Tail(Cons P Empty) =
- Isempty(Cons P Empty) =

Oefening 9:

- Head(Cons N(Cons P Empty)) =
- Tail(Cons N(Cons P Empty)) =
- Isempty(Cons N(Cons P Empty)) =

Extra Opgaven: Lambdarekenen:

Extra Oefening 1:

(a) (λx.(λy.(yx)))
(b) (λx.(λy.(λz.((xz)(yz)))))
(c) (((p(λx.(λy.(λz.((xz)(yz))))))p)q)

Extra Oefening 2:

(a) λxy.x(xy)
(b) λxyp.zpy(s(qy(xp)))

Extra Oefening 3:

(a) {y, x}
(b) {y}

Extra Oefening 4:

(a) (λx.xxx)(((λyz.y)I)(SS))= <β-conversie> (λx.xxx)((λz.I)(SS))
= <β-conversie> (λx.xxx)I
= <β-conversie> I I I
= <def I> ((λx.x)(λx.x))(λx.x)
= <β-conversie> (λx.x)(λx.x)
= <β-conversie> (λx.x)
= <def I> I

(b) (λyz.zy)((λx.xxx)(λx.xxx))(λp.I) = <β-conversie> (λz.z((λx.xxx)(λx.xxx)))(λp.I)
= <β-conversie> (λp.I)((λx.xxx)(λx.xxx))
= <β-conversie> I

Extra Oefening 5:

SKK = <def S, def K> ((λxyz.xz(yz))(λxy.x))(λxy.x)
= <β-conversie> (λyz.(λxy.x)z(yz))(λxy.x)
= <β-conversie> (λz.((λxy.x)z)((λxy.x)z))
= <β-conversie> (λz.(λy.z)((λxy.x)z))
= <β-conversie> (λz.(λy.z)(λy.z))
= <β-conversie> (λz.z)
= <α-conversie> (λx.x)
= <def I> I

Extra Oefening 6:

KI = <def K, def I> (λxy.x)(λx.x)
= <β-conversie> (λy.(λx.x))
= <haakjes> λyx.x
= <α-conversie> λxy.y

Extra Oefening 7:

(a) Not Tru = <def Not, def Tru> (λxyz.xzy) (λxy.x)
= <β-conversie> (λyz.(((λxy.x)z)y))
= <β-conversie> (λyz.((λy.z)y))
= <β-conversie> (λyz.z)
= <α-conversie> (λxy.y)
= <def Fls> Fls

(b) Not Fls = <def Not, def Fls> (λxyz.xzy)(λxy.y)
= <β-conversie> (λyz.(((λxy.y)z)y))
= <β-conversie> (λyz.((λy.y)y))
= <β-conversie> (λyz.y)

= <α-conversie> (λxy.x)
= <def Tru> Tru

(c) And Fls Fls = <def And, def Fls> ((λxy.(xy(λxy.y)))(λxy.y))(λxy.y)
= <β-conversie> (λy.((λxy.y)y(λxy.y)))(λxy.y)
= <β-conversie> ((λxy.y)(λxy.y))(λxy.y)
= <β-conversie> (λy.y)(λxy.y)
= <β-conversie> (λxy.y)
= <def Fls> Fls

(d) And Fls Tru = <def And, def Fls, def Tru> ((λxy.(xy(λxy.y)))(λxy.y))(λxy.x)
= <β-conversie> (λy.((λxy.y)y(λxy.y)))(λxy.x)
= <β-conversie> ((λxy.y)(λxy.x))(λxy.y)
= <β-conversie> (λy.y)(λxy.y)
= <β-conversie> (λxy.y)
= <def Fls> Fls

(e) And Tru Fls = <def And, def Tru, def Fls> ((λxy.(xy(λxy.y)))(λxy.x))(λxy.y)
= <β-conversie> (λy.((λxy.x)y(λxy.y)))(λxy.y)
= <β-conversie> ((λxy.x)(λxy.y))(λxy.y)
= <β-conversie> (λy.(λxy.y))(λxy.y)
= <β-conversie> (λxy.y)
= <def Fls> Fls

(f) And Tru Tru = <def And, def Tru, def Tru> ((λxy.(xy(λxy.y)))(λxy.x))(λxy.x)
= <β-conversie> (λy.((λxy.x)y(λxy.y)))(λxy.x)
= <β-conversie> ((λxy.x)(λxy.x))(λxy.y)
= <β-conversie> (λy.(λxy.x))(λxy.y)
= <β-conversie> (λxy.x)
= <def Tru> Tru

(g) Or Fls Fls = <def Or, def Fls, def Fls> (λxy.((x(λxy.x))y))(λxy.y))(λxy.y)
= <β-conversie> (λy.(((λxy.y)(λxy.x))y)))(λxy.y)
= <β-conversie> ((λxy.y)(λxy.x))(λxy.y)
= <β-conversie> (λy.y)(λxy.y)
= <β-conversie> (λxy.y)
= <def Fls> Fls

(h) Or Fls Tru = <def Or, def Fls, def Fls> (λxy.((x(λxy.x))y))(λxy.y))(λxy.x)
= <β-conversie> (λy.(((λxy.y)(λxy.x))y)))(λxy.x)
= <β-conversie> ((λxy.y)(λxy.x))(λxy.x)
= <β-conversie> (λy.y)(λxy.x)
= <β-conversie> (λxy.x)
= <def Tru> Tru

(i) Or Tru Fls = <def Or, def Fls, def Fls> (λxy.((x(λxy.x))y))(λxy.x))(λxy.y)
= <β-conversie> (λy.(((λxy.x)(λxy.x))y)))(λxy.y)
= <β-conversie> ((λxy.x)(λxy.x))(λxy.y)
= <β-conversie> (λy.(λxy.x))(λxy.y)

= <β-conversie> (λxy.x)
= <def Tru> Tru

(j) Or Tru Tru = <def Or, def Fls, def Fls> (λxy.((x(λxy.x))y))(λxy.x))(λxy.x)
= <β-conversie> (λy.(((λxy.x)(λxy.x))y))(λxy.x)
= <β-conversie> ((λxy.x)(λxy.x))(λxy.x)
= <β-conversie> (λy.(λxy.x))(λxy.x)
= <β-conversie> (λxy.x)
= <def Tru> Tru

Extra Oefening 8: