HW 1

1. let x=4+5 in (3*x)
   → 3*(4+5) — let reduction
   → 3*9 — arithmetic
   → 27 — arithmetic

2. (\x → 3*x) (4+5)
   ~~→ 3*(4+5))~~
   → 3*9 — lambda reduction
   → 27 — arithmetic

3. ((\x → (\y → x+(3*y))) 4)
   → (\y → 4+(3*y)) — lambda reduction
   4+(3*1) — lambda reduction
   4+3 — arithmetic
   7 — arithmetic

4. ((\x → (\y → x+(3*x))) 4)
   (\y → 4+(3*4)) — lambda reduction
   (\y → 4+(12) — arithmetic
   (\y → 16) — arithmetic
   16 — lambda reduction

5. ((\x → (\y → y+(3*y))) 4)
   (\y → y+(3*y) — lambda reduction
   1+(3*1) — lambda reduction
   1+(3) — arithmetic
   4 — arithmetic

6. Let x=4 in (let y=1 in (x+(3·y)))

Let x=4 in (x+(3·1)) — let reduction

Let x=4 in (x+(3)) — arithmetic

(4+(3)) — let reduction

7 — arithmetic

7. let x=4 in (let y=1+x in (x+(3·y)))

let x=4 in (x+(3·(1+x))) — let reduction

4+(3·(1+4)) — let reduction

4+(3·(5)) — arithmetic

4+(15) — arithmetic

19 — arithmetic

8. let x=4 in (let x=3 in (2+x))

let x=4 in (2+3) — let reduction

2+3 — let reduction

5 — arithmetic

9. (\y → y+((\y → 3·y) 4)) 5

\y → y+(3·4) 5 — lamda reduction

5+(3·4) — lambda reduction

5+(12) — arithmetic

17 — arithmetic

10. $\lambda y \to ((\lambda y \to 3 \cdot y)\ 4) + y)\ 5$

$(\lambda y \to (3 \cdot 4) + y)\ 5$ — lambda reduction

$(\lambda y \to (12 + y)\ 5$ — arithmetic

$(12 + 5)$ — lambda reduction

17 — arithmetic

11. $(\lambda x \to x \cdot (\text{let } x = 3 * 2 \text{ in } (x+7)) + x)\ 4$

$(\lambda x \to x \cdot (3 * 2) + 7) + x)\ 5\ 4$ — let reduction

$(\lambda x \to x \cdot ((6) + 7) + x) + 4$ — arithmetic

$(\lambda x \to x \cdot (13) + x)\ 4$ — arithmetic

$4 \cdot (13) + 4$ — lambda reduction

$52 + 4$ — arithmetic

56 — arithmetic

12. $g\ ((\text{let } x = 4 \text{ in } (\lambda y \to x + y))\ 2)$

$g\ ((\lambda y \to 4 + y)\ 2)$ — let reduction

$g\ (4 + 2)$ — lambda reduction

$g\ 6$ — arithmetic

$(\lambda z \to z + 4)\ 6$ — substitution

$6 + 4$ — lambda reduction

10 — arithmetic

#1

f = \s → case s of {Rock → 334; paper → 138;
scissors → 993}

13. let fn = \x → (let y = 3 in (x → x+y)) in fn 4

let fn = \x → (x+3) in fn 4

((\x → (x+3)) 4

4+3

7

let reduction
is substitution,
lambda reduction
arithmetic

14. let fn = (let y = 3 in \x → x+y) in fn 4

let fn = (\x → x+3) in fn 4

(\x → x+3) 4

4+3

7

is simplified more

99

let reduction
substitution
lambda reduction
arithmetic
substitution
case reduction

15. ~~f (let fn → fn Rock) (\x → what It Beats x))~~

~~f ((\x → what It Beats x) Rock~~

~~f (what It Beats Rock)~~

f (\s → case s of {Rock → scissors; paper → Rock;
scissors → paper}) Rock)

f (case Rock of {Rock → scissors; paper → Rock; scissors → Paper})

f (scissors)

(\s → case s of {Rock → 334; paper → 138; scissors → 993}) scissors

lambda reduction

substitution

lambda reduction

case reduction

15. f ((\fn → fn Rock) (\x → WhatItBeats x))

f((\x → WhatItBeats x) Rock — lambda reduction

f((WhatItBeat Rock)) — lambda reduction

f(\s → case s of { Rock → scissors; Paper → Rock; scissors → Paper }) Rock) — substitution

f(case Rock of { Rock → scissors; Paper → Rock; scissors → paper } — lambda red

f (scissors) — case reduction

(\s → case s of { Rock → 334; Paper → 138; scissors → 99 }) scissors — substitution

(case scissors of { Rock → 334; Paper → 138; scissors → 99 }) — lambda reduction

99 — case reduction

HW 1 16-20

16. ((f -> (\x -> f (f x))) WhatItBeats) Paper)

(\x -> WhatItBeats (WhatItBeats x)) Paper — lambda reduction

WhatItBeats (WhatItBeats Paper) — lambda reduction

(WhatItBeats ((\s -> case s of {Rock -> Scissors; Paper -> Rock; Scissors -> Paper}) Paper)) — substitution

WhatItBeats (case Paper of {Rock -> Scissors; Paper -> Rock; Scissors -> Paper}) — lambda reduction

WhatItBeats (Rock) — case reduction

((\s -> case s of {Rock -> Scissors; Paper -> Rock; Scissors -> Paper}) Rock) — substitution

case Rock of {Rock -> Scissors; Paper -> Rock; Scissors -> Paper} — lambda reduction

Scissors — case reduction

17. WhatItBeats (case Paper of {Rock -> Paper; Paper -> Rock; Scissors -> Scissors})

WhatItBeats Rock — case reduction

((\s -> case s of {Rock -> Scissors; Paper -> Rock; Scissors -> Paper}) Rock) — substitution

case Rock of {Rock -> Scissors; Paper -> Rock; Scissors -> Paper} — lambda reduction

Scissors — case reduction

16. (case (Win Rock) of {Draw → whatIfBeats; win z →
(case (win Rock) of {Draw → (\s → case s of {Rock → scissors;
Paper → Rock; scissors → paper}); win z → (\s →
scissors)}) Paper
(\s → scissors) paper
scissors

substitution
case reduction
lambda reduction
does nothing

19. (case (Win (WhatIfBeats Rock)) of {Draw → n; win x → (n + f x)};
(case (win (\s → case s of {Rock → scissors; Paper → Rock;
scissors → Paper}) Rock) of {Draw → n; win x →
(n + f x)}
substitution
(case (win (case Rock of {Rock → scissors; Paper → Rock;
scissors → Paper}) of {Draw → n; win x → (n + f x)}
substitution
(case (win scissors) of {Draw → n; win x → (n + f x)};
case reduction
substitution
(case (win scissors) of {Draw → 1; win x → (1 + f x)}
(1 + f scissors);
case reduction
(1 + (\s → case s of {Rock → 334; Paper → 138;
scissors → aa}) scissors)
substitution
lambda reduction
(1 + case scissors of {Rock → 334; Paper → 138;
scissors → aa}
1 + aa
100
case reduction
arithmetic

20) let y=2 in (case (win (whatItBeats Rock)) of
{Draw→n; win y→(n+f y)}+y)
let y=2 in (case (win (\s→case s of {Rock→scissors;
Paper→Rock; scissors→Paper}) Rock) of
{Draw→n; win y→(n+f y)}+y) substitution
lambda reduction
let y=2 in (case (win (case Rock of {Rock→scissors;
Paper→Rock; scissors→Paper}) of {Draw→n; win y
→(n+f y)}+y)
let y=2 in (case (win (scissors)) of {Draw→n; case reduction
win y→(n+f y)}+y)
let y=2 in (case (win (scissors)) of substitution
{Draw→1; win y→(1+f y)}+y
let y=2 in ((1+f (scissors))+y) case reduction
substitution
(1+f (scissors))+2) let reduction
substitution
(1+ (\s→case s of {Rock→334; Paper→138;
scissors→993)) scissors)+2)
(1+case scissors of {Rock→334; Paper→138; lambda reduction
scissors→993))+2)
(1+(99))+2) case reduction
100+2 arithmetic
102 arithmetic