Russel's Paraclox

$\{1, 2, 3\}$ $\{1.5\}$ $\{\{\{\{\ldots\}\}\}\}$

$\{\{1\}, \{1, 2\}\}$

R = all sets which contain themselves

$R \in R$?

yes

if $R \in R$ then $R \in R$.

if $R \notin R$ then $R \notin R$

P = all sets which do not contain themselves

$P \in P$?

if $P \in P$ then $P \notin P$

if $P \notin P$ then $P \in P$

Function element → element = 1 Or element = 4 → {1, 4}

Function element → element > 0

{ {1, 2}, {1}, {1, 1}, ... }

Function element → element 1 ← set of all sets containing 1

Function element → element element R

(Function e → e e) (Function e → e e) ω-combinator

(Function e → e e) (Function e → e e)

(Function e → Nof (e e))

Let sum' = Function self → Function n →
If n = 1 Then 1 Else
self self (n-1) + n

e_1: sum'

In
Let sum = sum' sum' In
sum 3

e_2

Let sum = sum' sum' In
sum 3

sum' sum' 3

Application Rule: sum': self is sum'

(Function n → If n = 1 Then 1 Else
sum' sum' (n-1) + n) 3

Function self → Function n →
If n = 1 Then 1 Else
self self (n-1) + n

sum'

If 3 = 1 Then 1 Else
sum' sum' (3-1) + 3

sum' sum' (3-1) + 3

Let sum' = ... In
Let mystery = Function self → (Function n →
If n = 1 Then 1 Else
1 + sum' self (n-1))

In
mystery mystery 5

sum' mystery 4 + 1

mystery mystery 3 + 4 + 1

sum' mystery 2 + 1 + 4 + 1

mystery mystery 1 + 2 + 1 + 4 + 1

1 + 2 + 1 + 4 + 1

```
Let rsum' = Function recurse -> Function n ->
              If n = 1 Then 1 Else
                recurse (n-1) + n
In
Let recurser' = Function self -> Function n ->
                  rsum' (self self) n
In
Let recurser = recurser' recurser'
In
Let rsum = rsum' recurser In
rsum 3
```

do recursion work
call rsum'
function of 1 int parameter

```
Function f ->
  Let ycr' = Function self -> Function x ->
               f (self self) x
  In
  Let ycr = ycr' ycr' In
  Function x -> f ycr x
```