SUBSTITUTION MODEL

Kind of EXPRESSIONS

- NUMBERS
- SYMBOLS
- LAMBDA EXPRESSIONS
- DEFINITIONS
- CONDITIONALS
- COMBINATIONS

RULE

TO EVAL AN APP

1. EVAL operator → procedure
2. EVAL operands → arguments
3. Apply procedure → args
   a. Copy the body of the procedure substituting the args supplied for the formal params of the procedure
   b. Eval the resulting new body

(IF Don't understand something... Follow this rule)

Sum of squares

```
(SOS 3 4)
(DEFINE (SOS X Y)
  (+ (SQ X) (SQ Y)))
(DEFINE (SQ X)
  (* X X))
```

EVALUATION ⇒ APPLICATION

```
(SOS 3 4)
(+ SOS (SQ 3) (SQ 4))
(+ (SQ 3) (* 4 4))
(+ (* 3 3) (* 4 4))
(+ 9 16)
25
```

RULE FOR CONDITIONALS

(IF <predicate>
<consequent>
<alternative>)

RULE

To Evaluate IF Expression

1. EVAL pred. EXP
   IF it yields TRUE
      evaluate the consequent
   otherwise
      eval the alt exp.

```
(DEFINE (+ X Y)
  (IF (= X 0)
      Y
      (+ (-1+ X) (1+ Y))))
```

Decrement Increment

```
(+ 3 4)
(IF (= 3 0) 4 (+ (-1+ 3) (1+ 4)))
(+ (-1+ 3) (1+ 4))
(+ (-1+ 2) (5)
(+ 2 5) . . . (+ 1 6) . . . (+ 0 7)
```

⇒ 7

PEANO ARITH
2 ways to Add whole #s

```
(DEFINE (+ X Y)
   (IF (= X 0)
       Y
       (+ (- 1+ X) (1+ Y))))
```

-OR-

```
(DEFINE (+ X Y)
   (IF (= X 0)
       Y
       (1+ (+ (-1+ X) Y))))
```

```
(+ 3 4)
(+ 2 5)
(+ 1 6)
(+ 0 7) => 7
```

time Expand ↓

```
(+ 3 4)
(1+ 2 4)
(1+ (1+ (+ 1 4)))
(1+ (1+ (1+ (+ 0 4))))
(1+ (1+ (1+ 4)))
(1+ (1+ 5))
(1+ 6) => 7
```

Contract ↑

Shape
order x
time = O(x)
Iteration
↓
space = O(1)

Linear Recursion
time = O(x)
space = O(x)

space

DIFFERENCE BETWEEN ITERATION & RECURSION

ITERATION IS A SYSTEM THAT HAS ALL OF ITS STATE IN EXPLICIT VARIABLES

RECURSION HAS ITS STATE WITH VARIABLES BUT ALSO OTHER THINGS UNDERNEATH

FIBONACCI:

0 1 1 2 3 5 8 13 21 34 55

♡ 1 2 3 4 . . . .

```
(DEFINE (FIB N)
  (IF (< N 2)
      N
      (+ (fib (- n 1))
         (fib (- n 2)))))
```

HANOI TOWER FROM

```
(DEFINE (MOVE N TO SPARE)
  ((OND ((=NO) "DONE"))
    (ELSE
      (MOVE (-H\ N) FROM FROM
      (PRINT-MOVE FROM TO
      (MOVE (-H N) SPARE TO FROM
```

— — — 4 123)

MOVE

(MOVE 3 132

(MOVE 2 123 ETC,

ETC.