

## Regular Expressions

regular expr  $r ::= c \mid \epsilon$

Single character  $\swarrow$   
Empty string  $\nwarrow$

$\mid r_1 \cdot r_2$  Concatenation

$\mid r_1 + r_2$  Alternatives  
Union

$\mid r^*$  Kleene-\*

Question: Given a regular expression  $r$   
given a string  $\sigma$   
does  $\sigma$  "match"  $r$ ?

Example :  $r = \text{Hi} \mid \text{Hello}$

$\sigma = \text{"Hello"}$

Example :  $r = \text{Hello}$

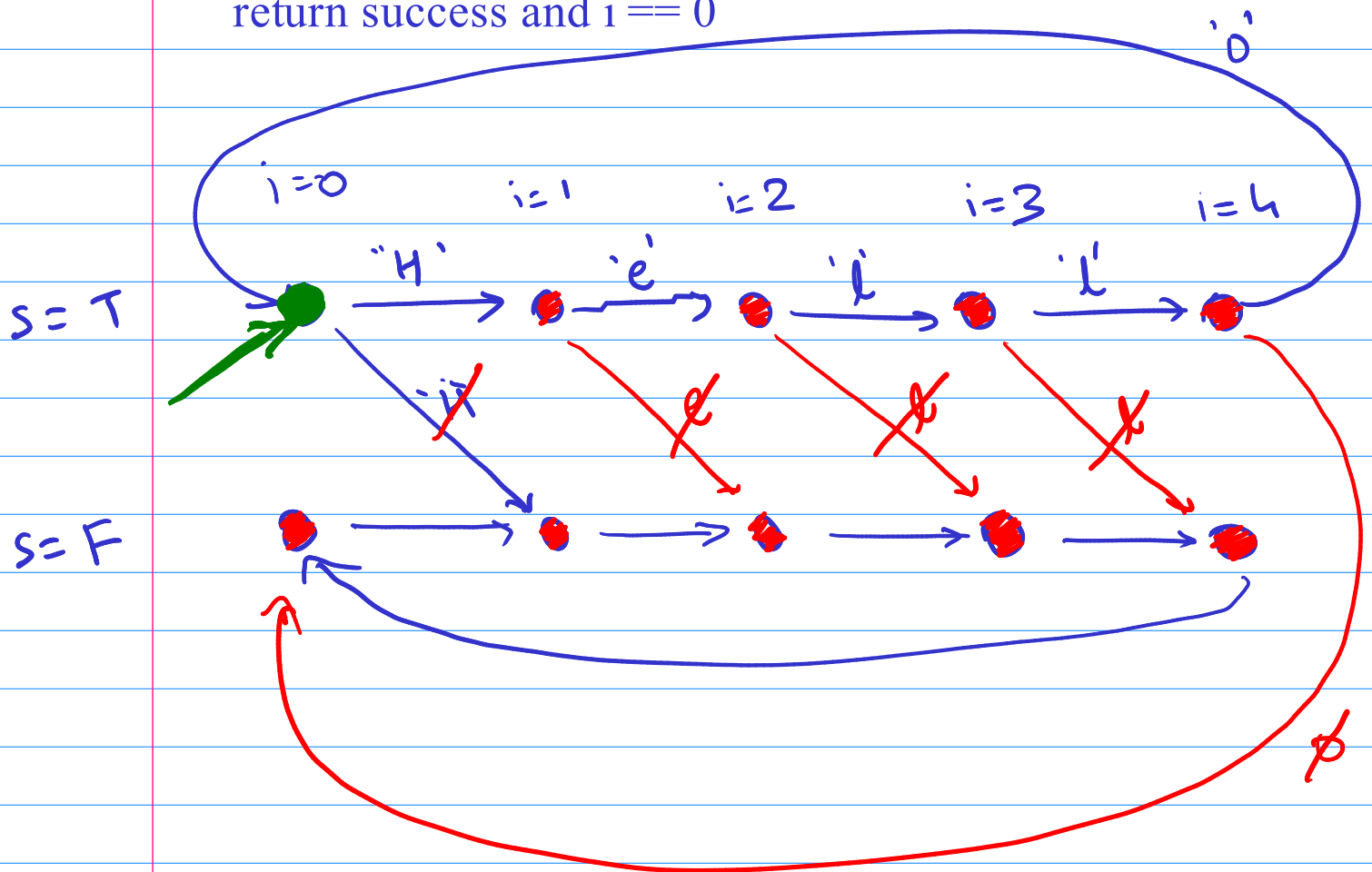
$\sigma = \text{"Help"}$

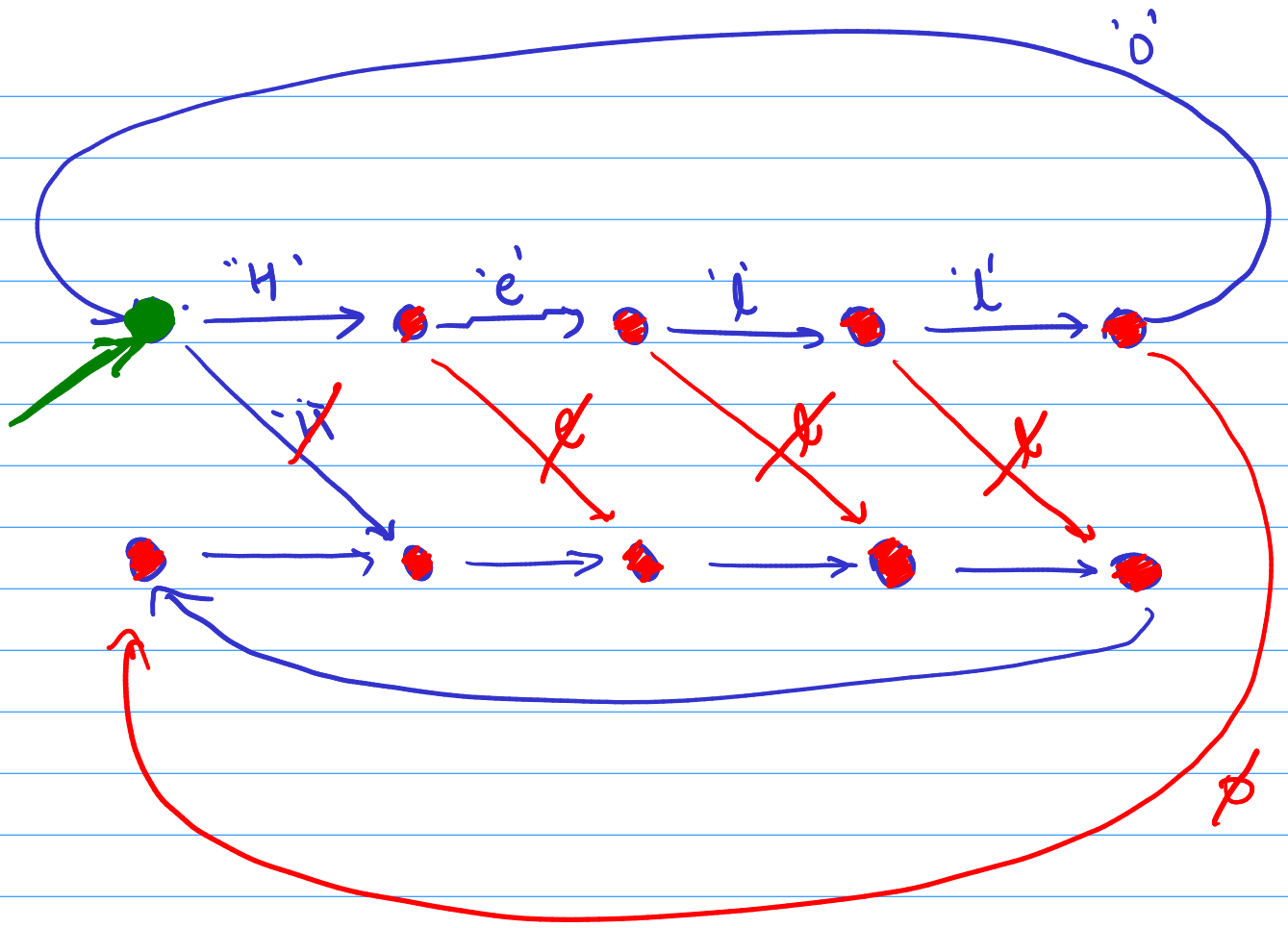
Example :  $r = \text{"Hello"}^*$

```

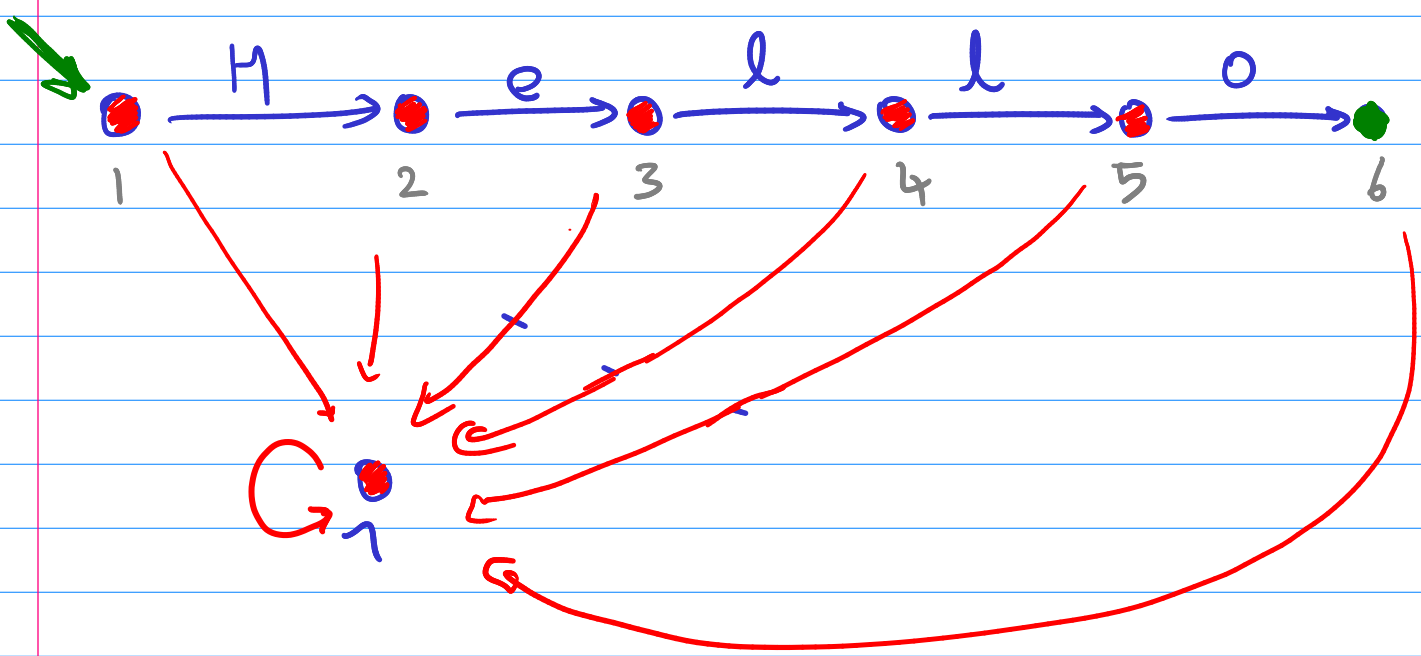
def matchesHelloStar(w):
    success = True
    i = 0
    for c in w:
        if c != 'Hello'[i]: success = False
        i = (i + 1) % len('Hello')
    return success and i == 0

```

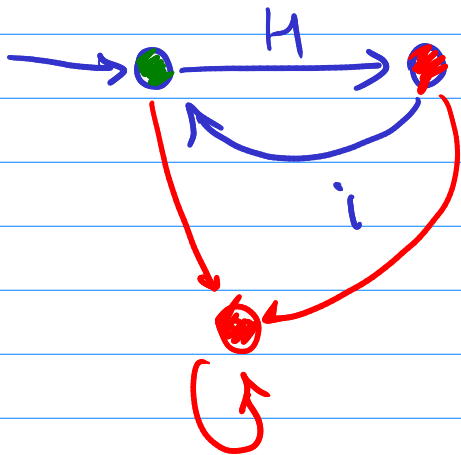




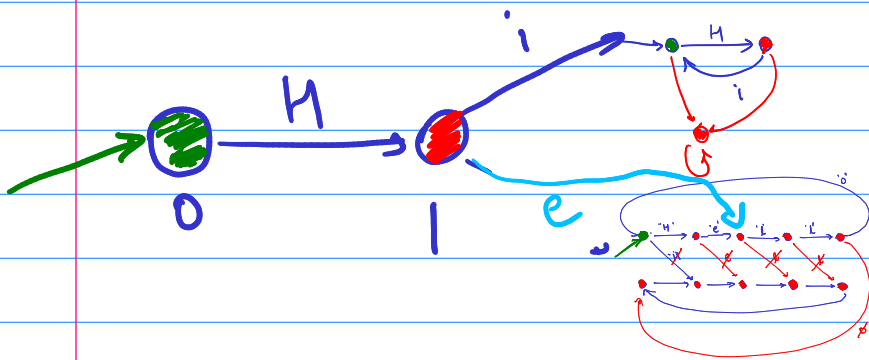
Example  $r = \text{'Hello'}$



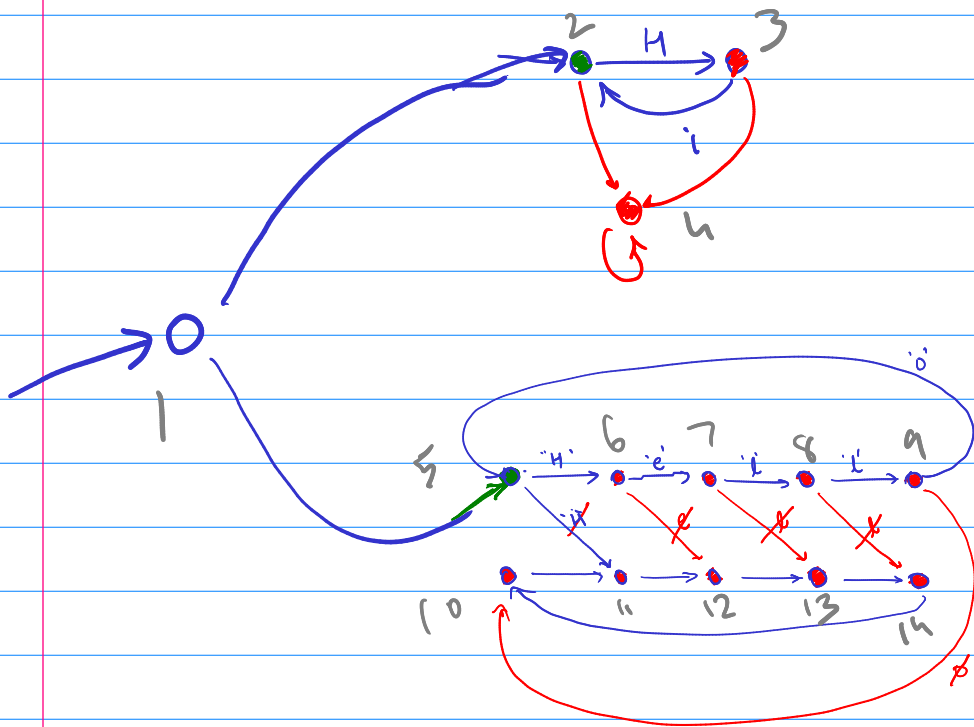
Example  $r = "Hi"*$



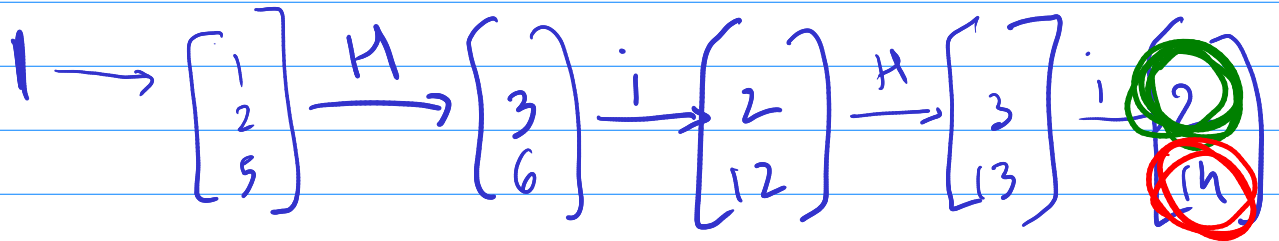
Example  $r = "Hi"* + "Hello"*$



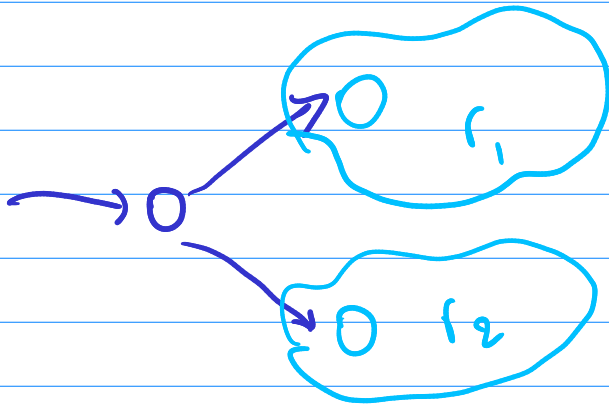
Kleene's Theorem



H: H:

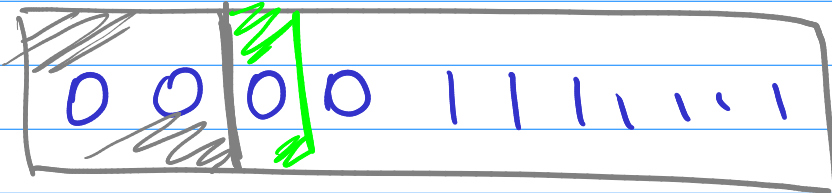


$r ::= c \mid \varepsilon \mid f$   
 |  $r_1 \cdot r_2$     Con  
 |  $r_1 + r_2$     Alt  
                    $\cup_n$   
 |  $r^*$             Kl

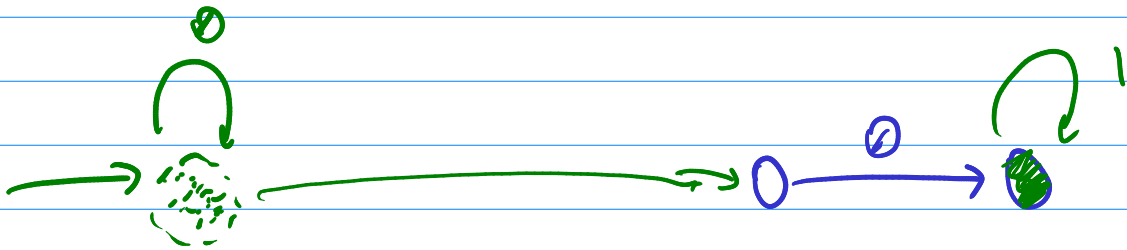
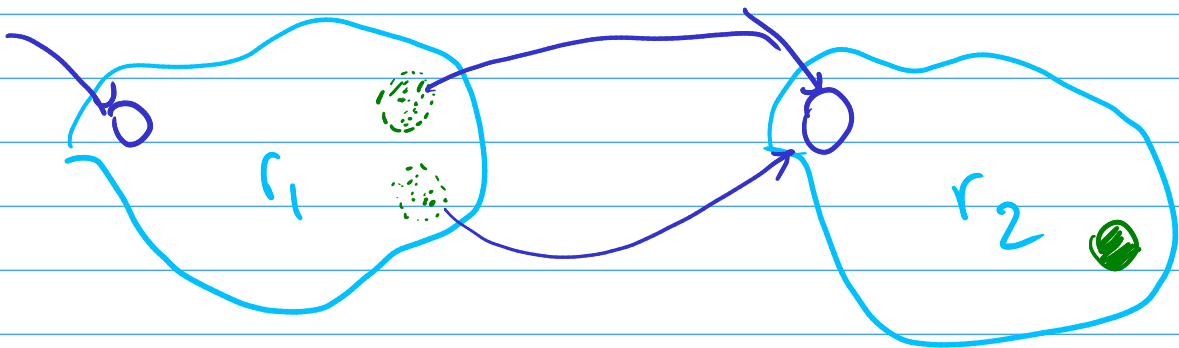


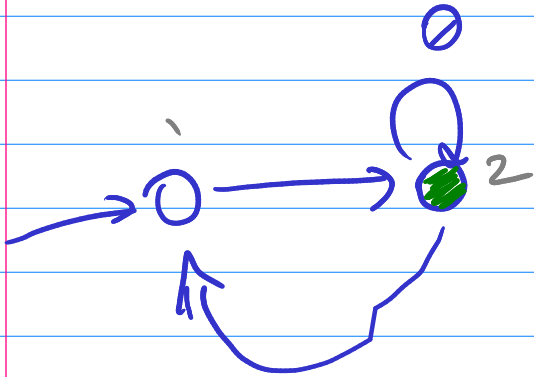
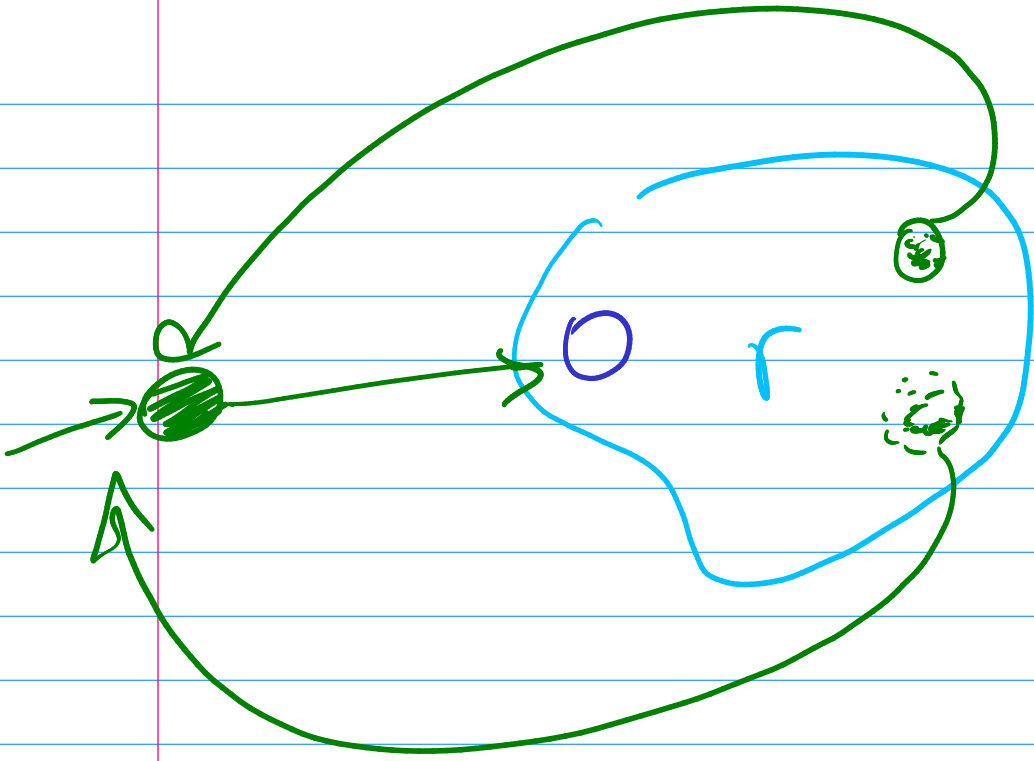


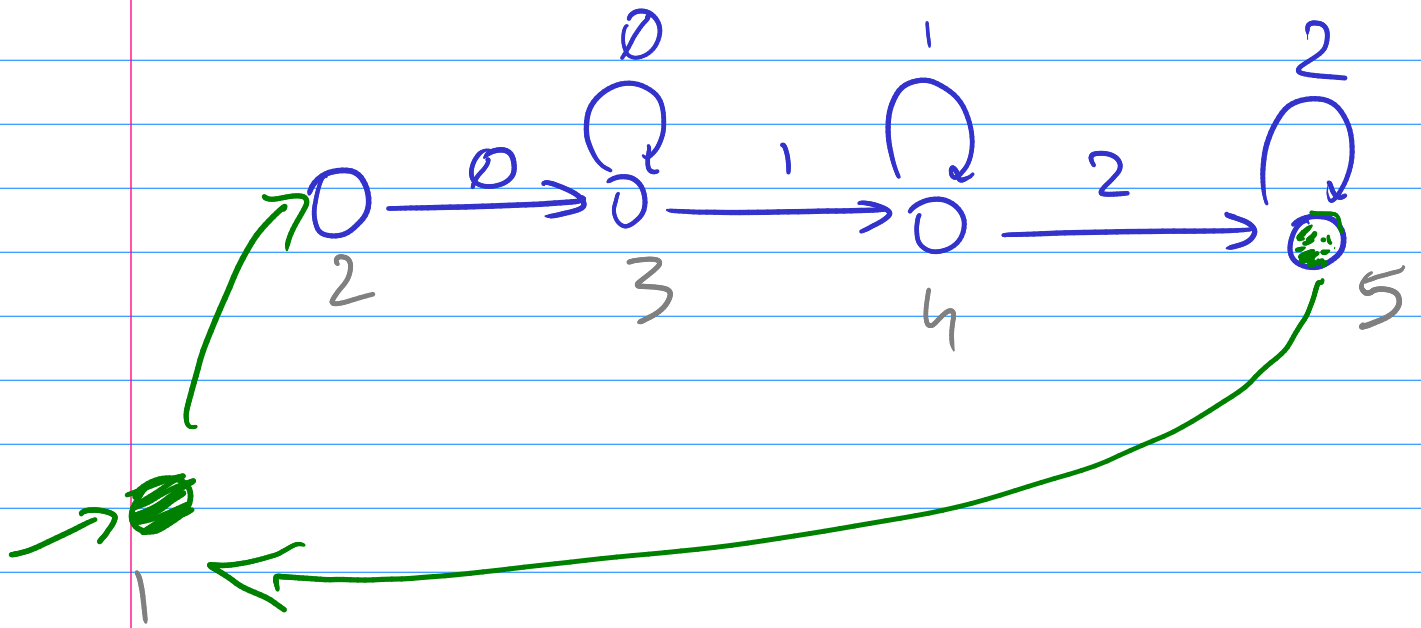
Example :  $r = 0^* \cdot (0 \cdot 1^*)$



$r_1 \cdot r_2$







# Types

$$3 + 5 \Rightarrow^* 8$$

$3 + "x"$



Runtime error

Type

error.

---

Language  $L_1$  (Untyped expressions)

$\text{expr} ::= \underbrace{0 \mid 1 \mid 2 \mid \dots}_{\text{Int}} \mid \underbrace{\text{true} \mid \text{false}}_{\text{Booleans}}$

$\mid \text{expr} + \text{expr} \mid \text{expr} - \text{expr}$

$\mid \text{expr} \leq \text{expr} \mid \text{expr}_1 \ \&\& \ \text{expr}_2$

$\mid \text{expr}_1 \ \parallel \ \text{expr}_2 \mid \text{not } \text{expr}$

$\mid \text{if } \text{expr} \text{ then } \text{expr}_1 \text{ else } \text{expr}_2$

Language  $L_2$  (Distinguished ints & bools)

$aexp ::= 0 \mid 1 \mid 2 \mid \dots$

$\mid aexp_1 + aexp_2 \mid aexp_1 - aexp_2$

$\mid \text{if } bexp \text{ then } aexp_1 \text{ else } aexp_2$

$bexp ::= \text{true} \mid \text{false}$

$\mid aexp_1 \leq aexp_2$

$\mid bexp_1 \ \&\ \ bexp_2 \mid bexp_1 \ \|\ \ bexp_2$

$\mid \text{not } bexp$

Language  $L_3$  (Ints, Booleans, Lists)

expr  $e ::= 0 \mid 1 \mid 2 \mid \dots \mid \text{true} \mid \text{false}$   
 $\mid [e_1 \ e_2 \ e_3 \ \dots \ e_x]$   
 $\mid e_1 + e_2 \mid e_1 - e_2$   
 $\mid e_1 \leq e_2 \mid \text{if } e_1 \text{ then } e_2 \text{ else } e_3$   
 $\mid e_1 @ e_2$   
 $\mid e_1 \llbracket e \rrbracket$

