

# FOREST: An Interactive Multi-tree Synthesizer for Regular Expressions

---

Margarida Ferreira, Miguel Terra-Neves, Miguel Ventura,  
Inês Lynce, Ruben Martins

# Date input field

---

Date:

Submit

Desired format:

DD/MM/YYYY

Regular expression:

`[0-9]{2}/[0-9]{2}/[0-9]{4}`

# DD/MM/YYYY format examples

---



19/08/1996

26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015



19/08/96

26-10-1998

22.09.2000

1/12/2001

29/9/2003

2015/08/31

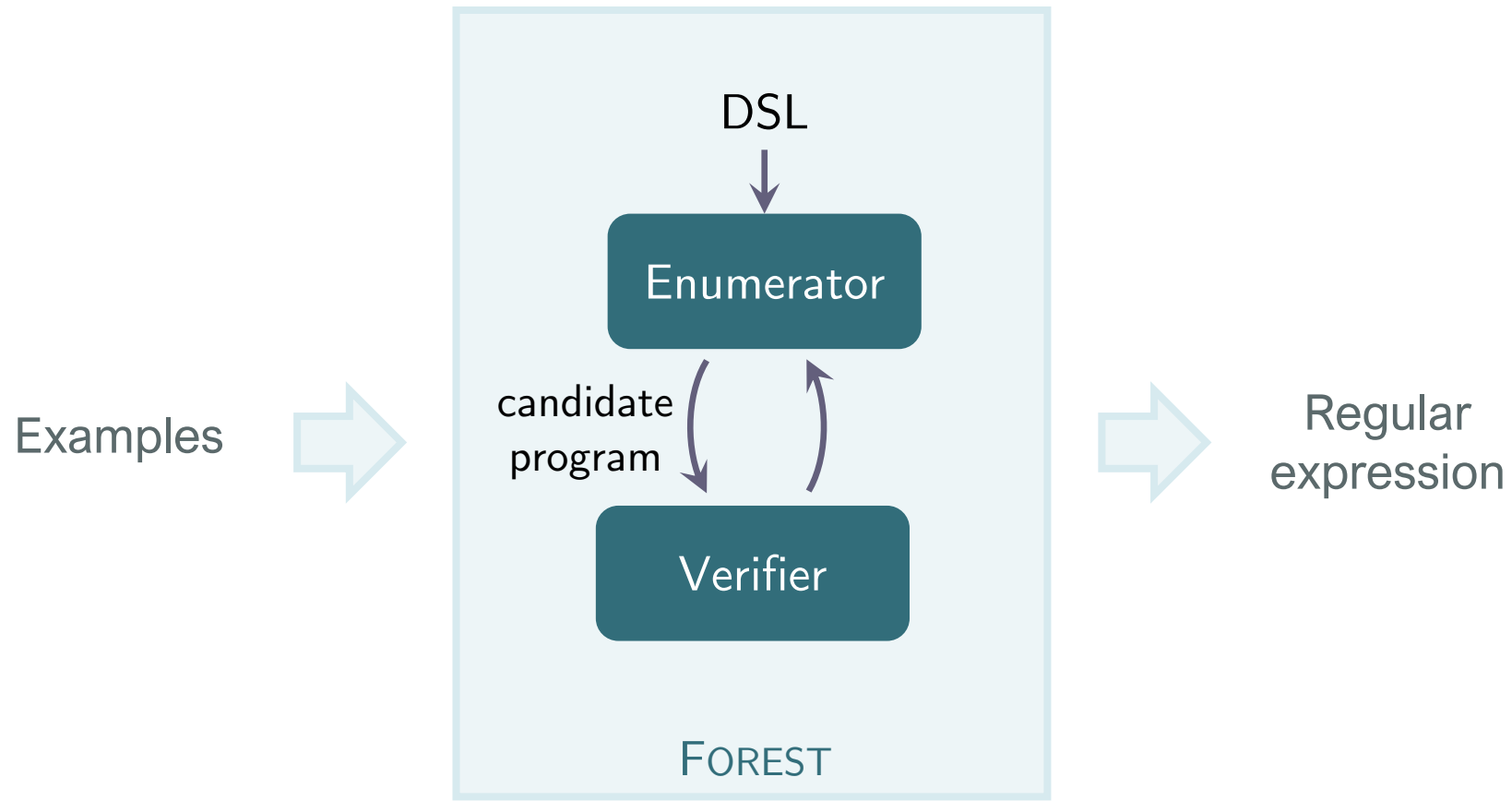
# Programming by Example

---



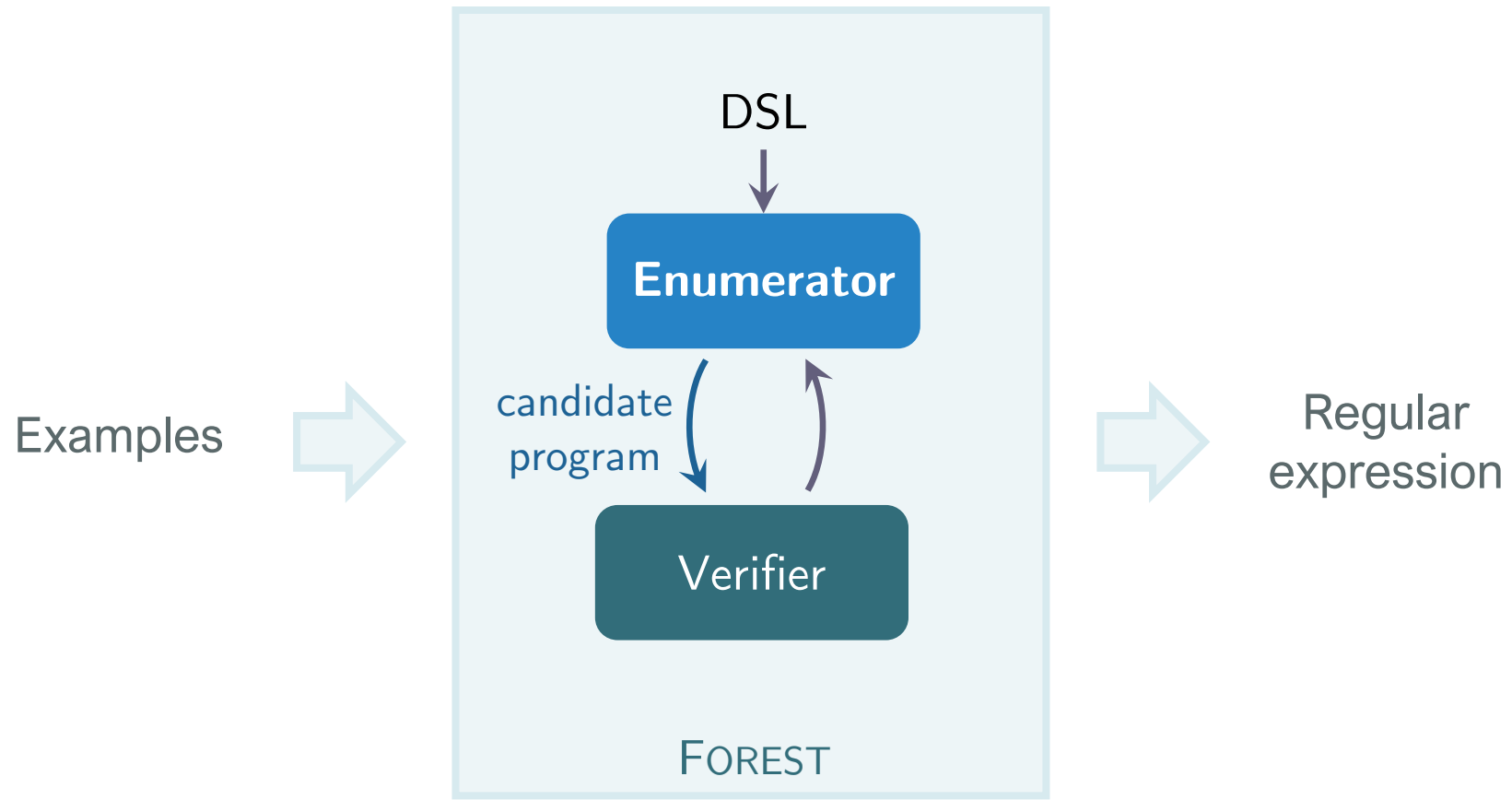
# Enumerative Search

---

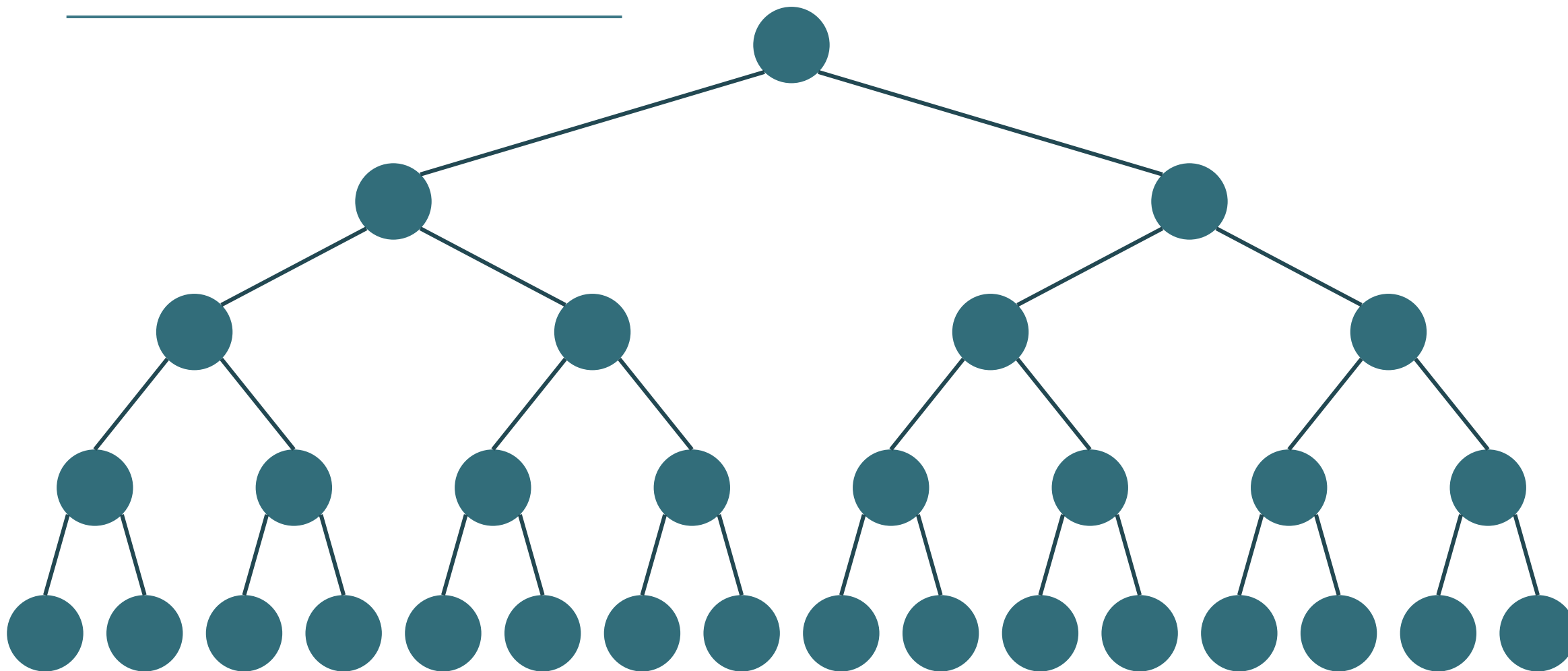


# Enumerative Search

---



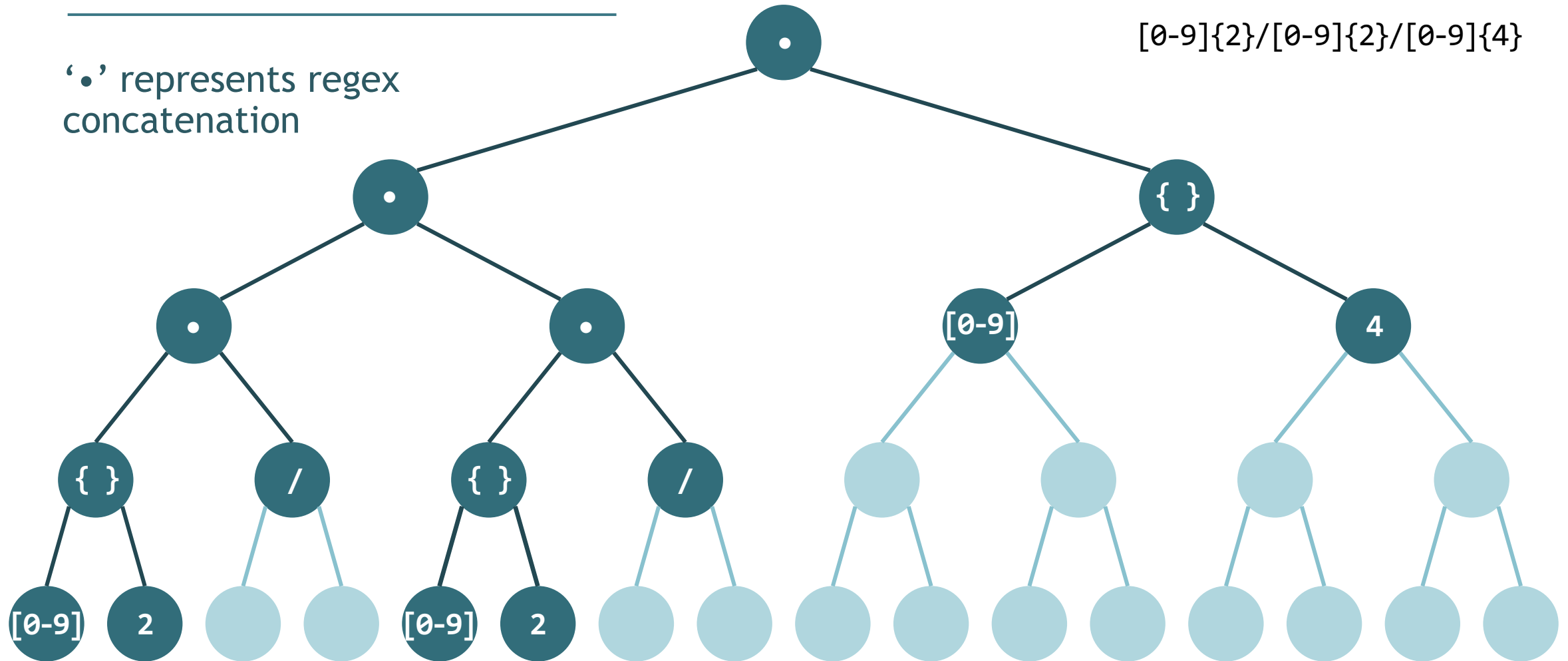
## *k*-trees



## Enumeration

### *k*-trees

‘•’ represents regex concatenation





Enumeration

## Multi-tree

---



19/08/1996

26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015

Valid examples

Enumeration

## Multi-tree

---

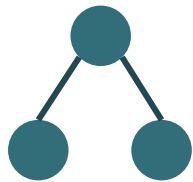
19	/	08	/	1996
26	/	10	/	1998
22	/	09	/	2000
01	/	12	/	2001
29	/	09	/	2003
31	/	08	/	2015

## Enumeration

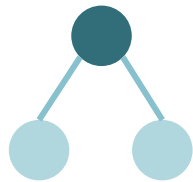
### Multi-tree

---

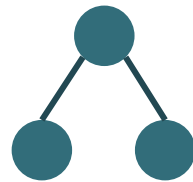
19	/	08	/	1996
26	/	10	/	1998
22	/	09	/	2000
01	/	12	/	2001
29	/	09	/	2003
31	/	08	/	2015



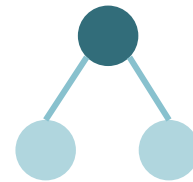
`[0-9]{2}`



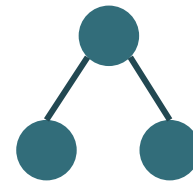
`/`



`[0-9]{2}`



`/`



`[0-9]{4}`

Enumeration

## Multi-tree

---

19/08/1996

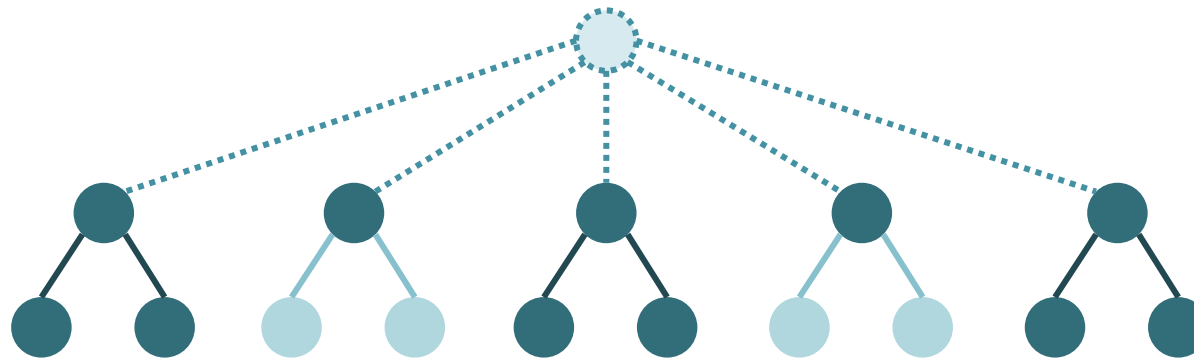
26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015

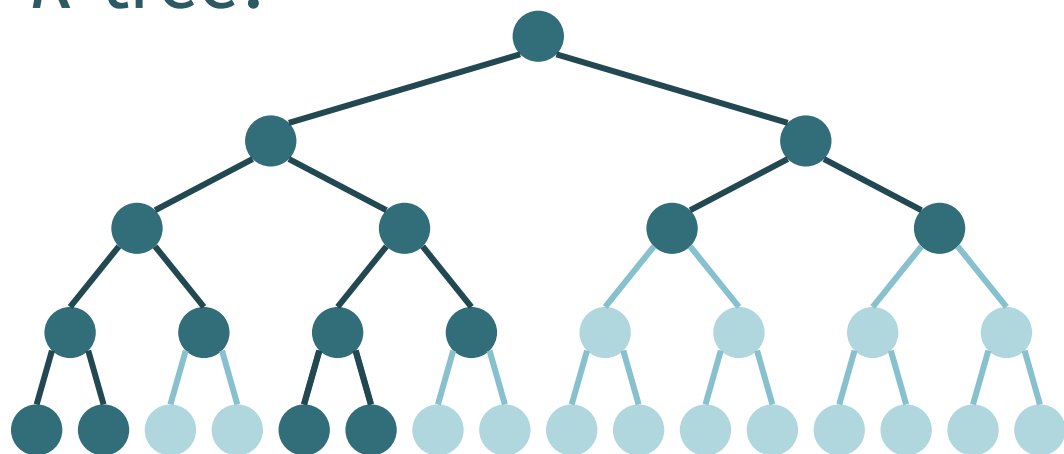


$[0-9]\{2\}/[0-9]\{2\}/[0-9]\{4\}$

# Enumeration

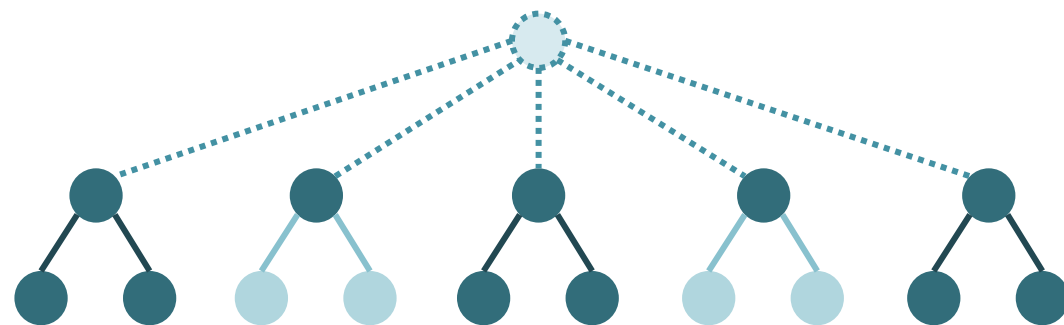
## K-tree vs. Multi-tree for $[0-9]\{2\}/[0-9]\{2\}/[0-9]\{4\}$

## *K*-tree:



- 31 nodes
  - 16 empty

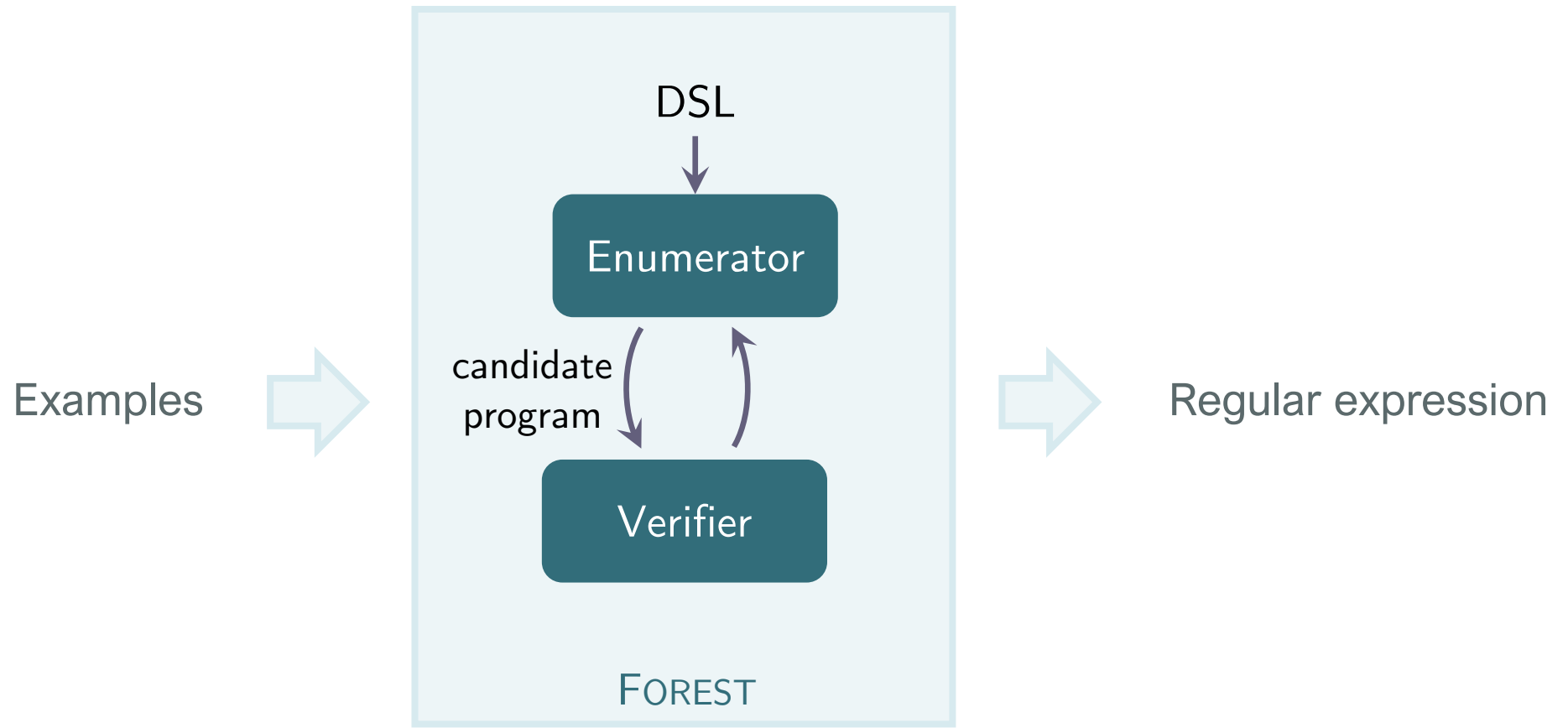
## Multi-tree:



- 15 (+1) nodes
  - 4 empty

# Enumerative Search

---



# Solution(s)?

---

$[0-9]\{2\}/[0-9]\{2\}/[0-9]\{4\}$

$[0-9]\{2\}/[0-9]\{2\}/[0-9]\{3,4\}$



19/08/1996

26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015



19/08/96

26-10-1998

22.09.2000

1/12/2001

29/9/2003

2015/08/31

# Solution(s)?

---

$[0-9]\{2\}/[0-9]\{2\}/[0-9]\{4\}$   
 $[0-9]\{2\}/[0-9]\{2\}/[0-9]\{3,4\}$

24/04/404





# Solution(s)?

---

`[0-9]{2}/[0-9]{2}/[0-9]{4}`

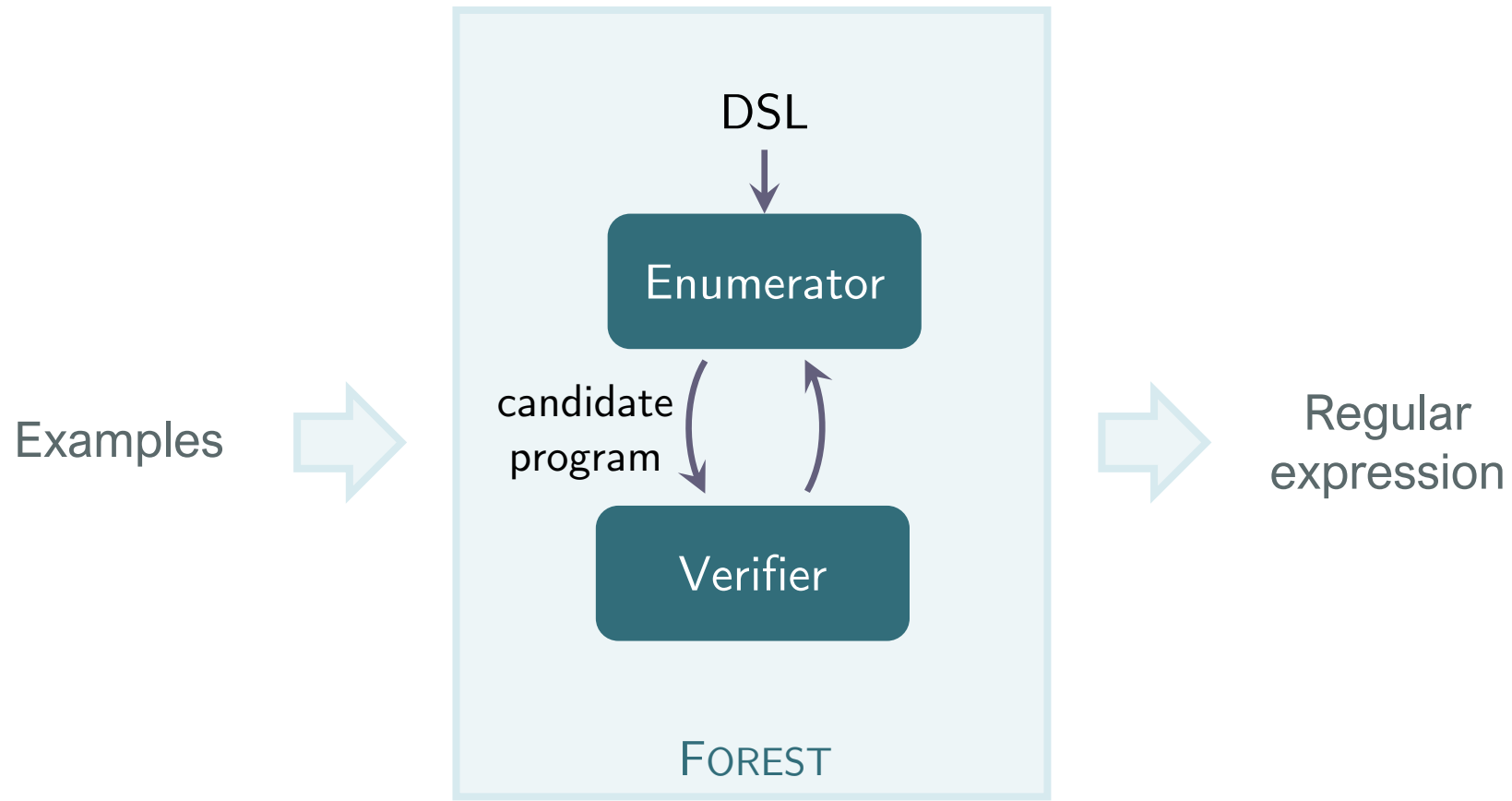
~~`[0-9]{2}/[0-9]{2}/[0-9]{3,4}`~~

24/04/404

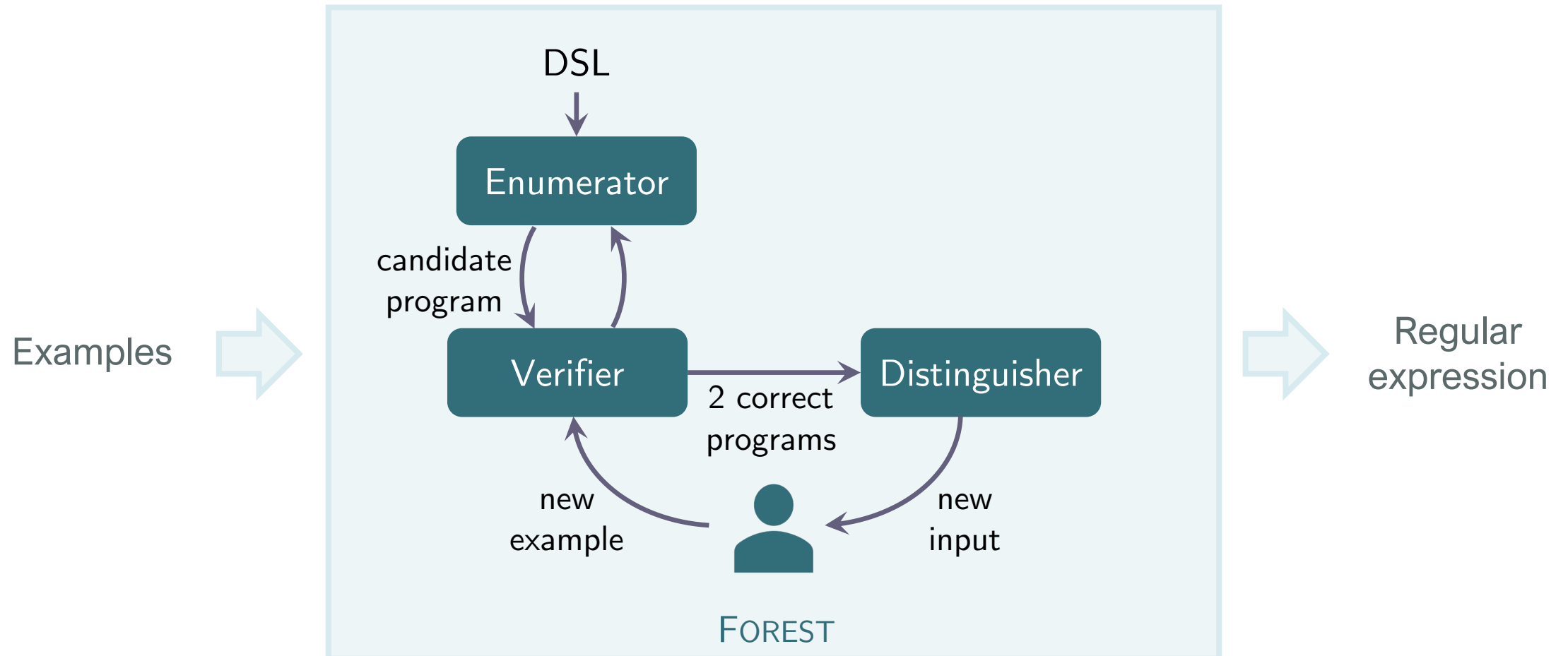


# Enumerative Search

---



# Interactive Enumerative Search



# Solution

---

$[0-9]\{2\}/[0-9]\{2\}/[0-9]\{4\}$



19/08/1996

26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015



19/08/96

26-10-1998

22.09.2000

1/12/2001

29/9/2003

2015/08/31

24/04/404

33/08/1996

26/00/1998

input's *format*? ✓

input's *values*? ✗

- $1 \leq \text{day} \leq 31$

- $1 \leq \text{month} \leq 12$

# Conditional invalid examples

---



19/08/1996  
26/10/1998  
22/09/2000  
01/12/2001  
29/09/2003  
31/08/2015





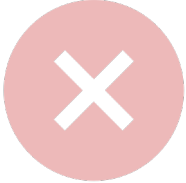
33/08/1996  
26/00/1998  
22/13/2000  
00/12/2001  
12/31/2003  
52/03/2015



19/08/96  
26-10-1998  
22.09.2000  
1/12/2001  
29/9/2003  
2015/08/31  
24/04/404

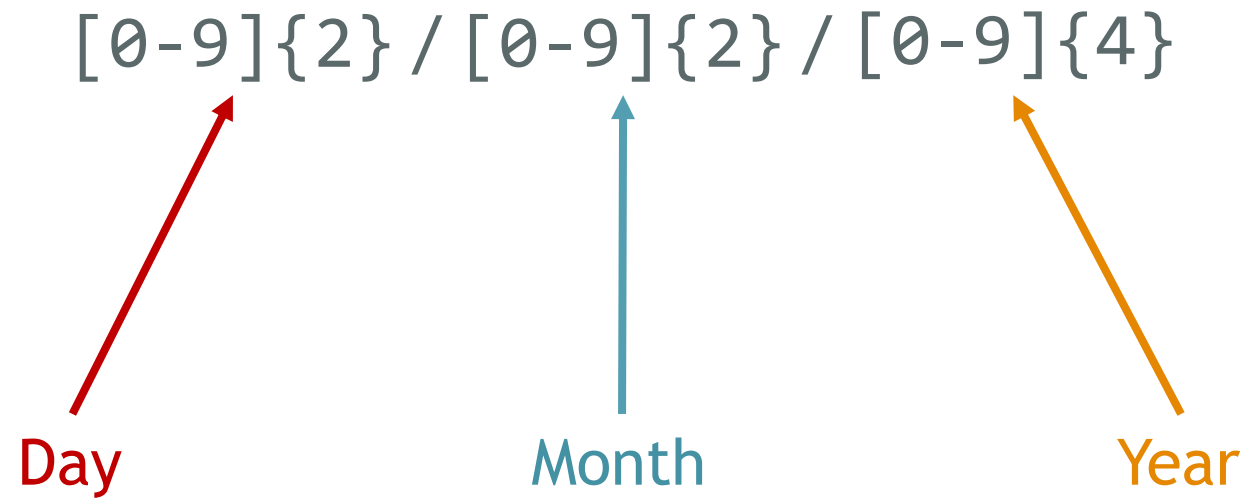
# Conditional invalid examples

---

			
Regular expression	✓	✓	X
Capture conditions	✓	X	-

# Capturing groups

---



# Capturing groups

---

`([0-9]{2})/[0-9]{2}/[0-9]{4}`

19/08/1996

→

19

`[0-9]{2}/([0-9]{2})/[0-9]{4}`

19/08/1996

→

08

`([0-9]{2})/([0-9]{2})/[0-9]{4}`

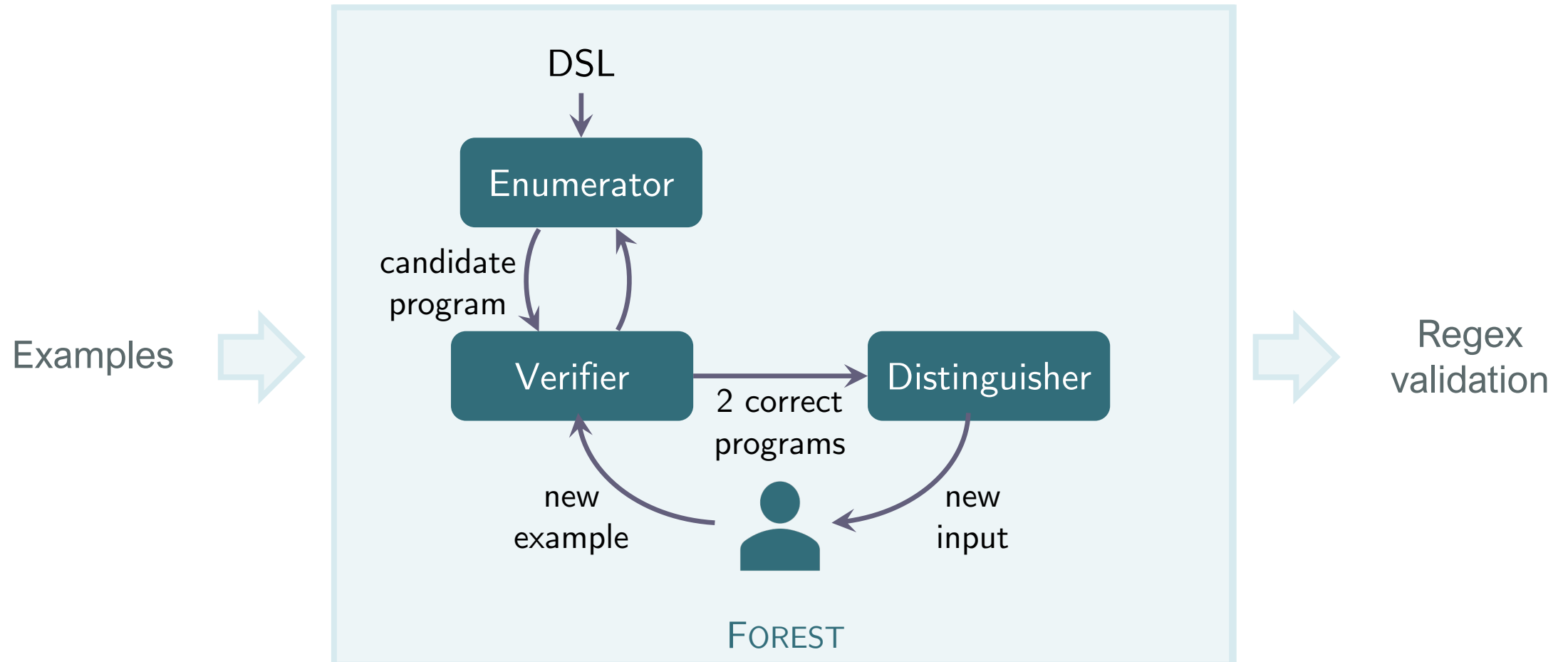
19/08/1996

→

`$0 = 19 ; $1 = 08`

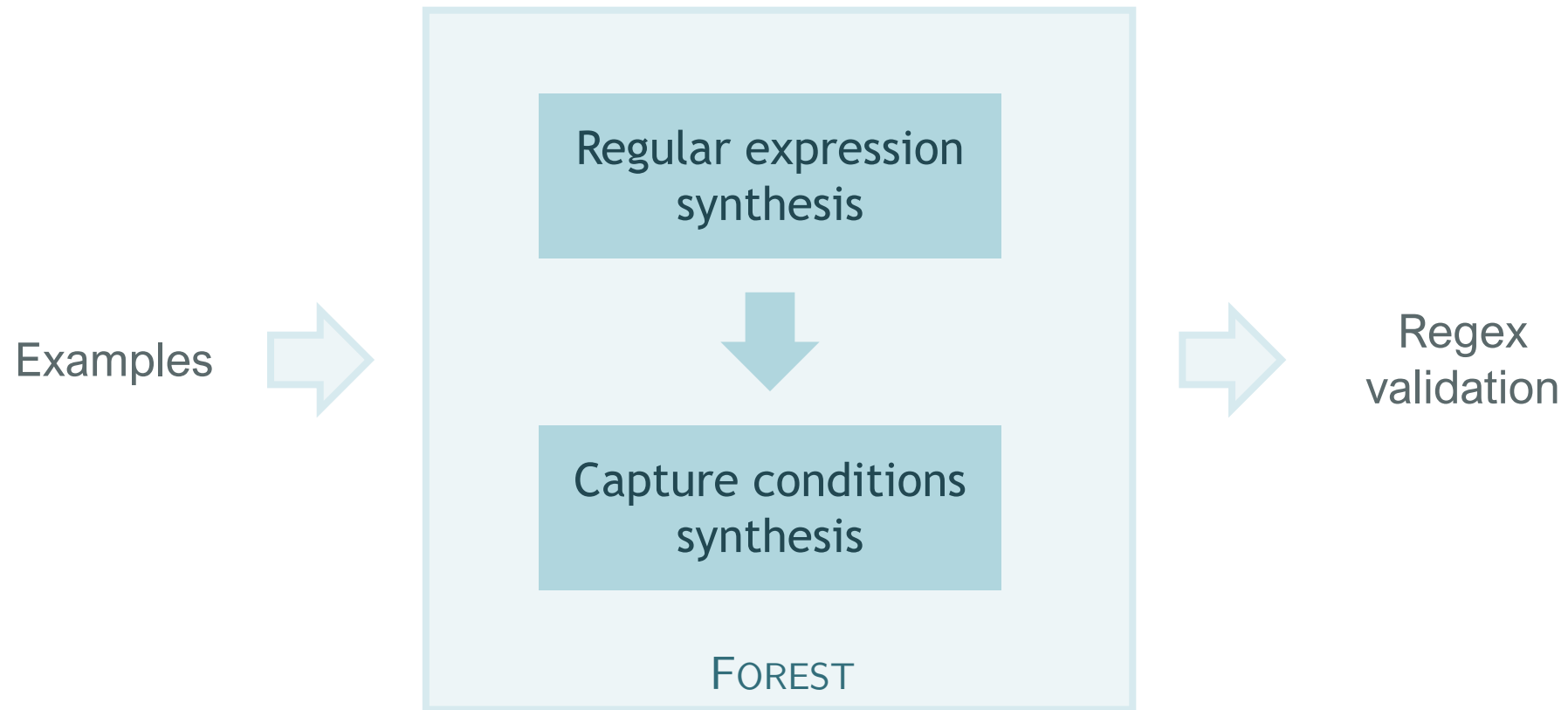


# Interactive Enumerative Search



# Interactive Enumerative Search

---



# Capturing groups enumeration

---

`[0-9]{2}/[0-9]{2}/[0-9]{4}`

# Capturing groups enumeration

---

`([0-9]{2})/([0-9]{2})/[0-9]{4}`

# Conditions over captures

---

$([\text{0-9}]\{2\})/([\text{0-9}]\{2\})/[\text{0-9}]\{4\}$

$\$0 \geq 1$  ;  $\$0 \leq 31$  ;  $\$1 \geq 1$  ;  $\$1 \leq 12$



19/08/1996

26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015



33/08/1996

26/00/1998

22/13/2000

00/12/2001

12/31/2003

52/03/2015

# Solution(s)?

---

$([0-9]\{2\})/([0-9]\{2\})/[0-9]\{4\}$

$\$0 \geq 1$  ;  $\$0 \leq 31$  ;  $\$1 \geq 1$  ;  $\$1 \leq 12$

$\$0 \geq 1$  ;  $\$0 \leq 32$  ;  $\$1 \geq 1$  ;  $\$1 \leq 12$



19/08/1996

26/10/1998

22/09/2000

01/12/2001

29/09/2003

31/08/2015



33/08/1996

26/00/1998

22/13/2000

00/12/2001

12/31/2003

52/03/2015

# Solution

---

$([0-9]\{2\})/([0-9]\{2\})/[0-9]\{4\}$

$\$0 \geq 1 ; \$0 \leq 31 ; \$1 \geq 1 ; \$1 \leq 12$

~~$\$0 > 1 ; \$0 \leq 32 ; \$1 \geq 1 ; \$1 \leq 12$~~

32/08/1996 

# FOREST's Synthesis

---

Examples



**FOREST**



Regex validation



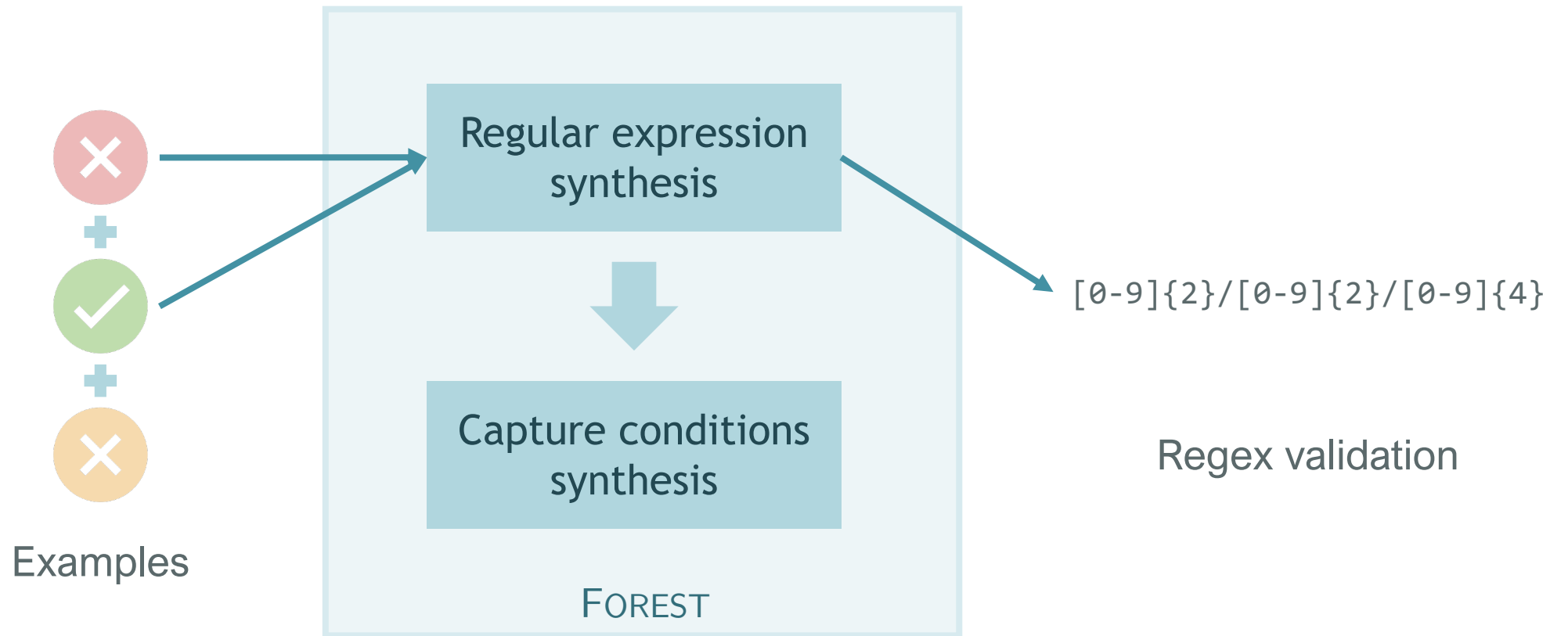
# FOREST's Synthesis

---

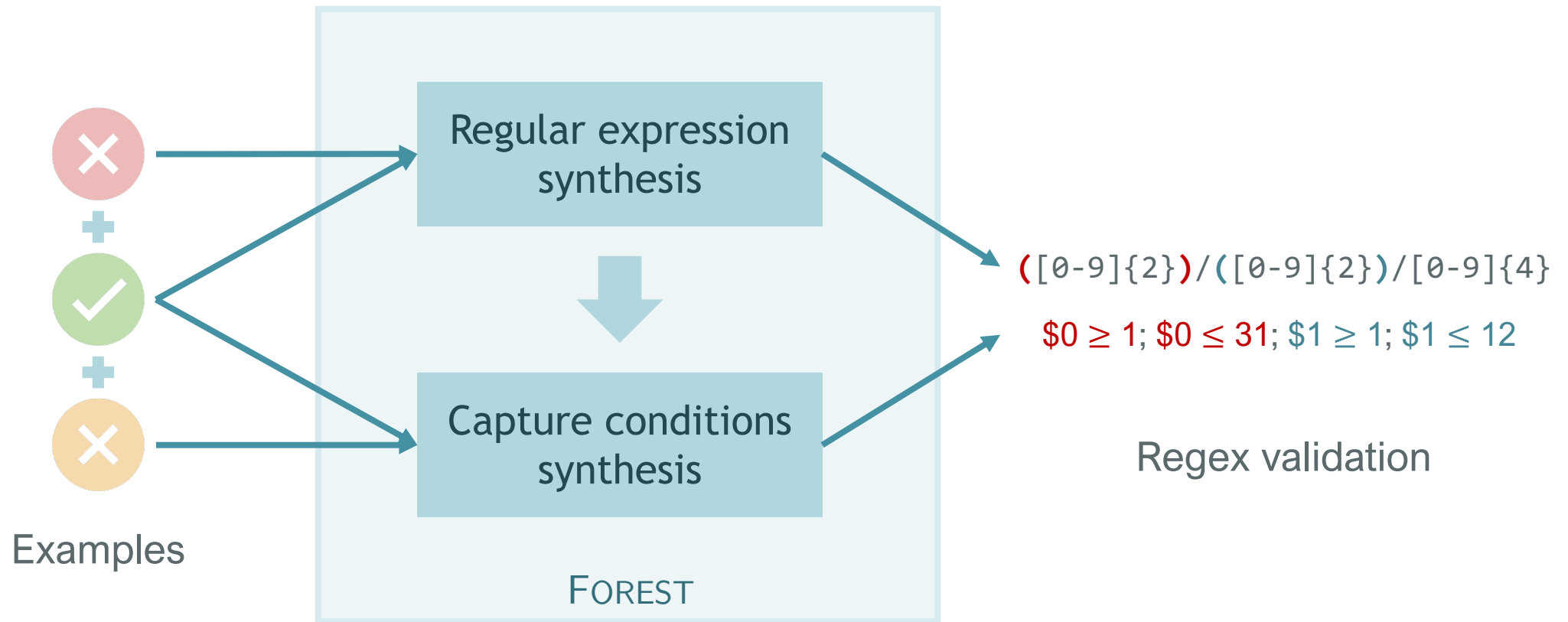


# FOREST's Synthesis

---



# FOREST's Synthesis



# Experimental results

# Benchmarks

---

64 regular expression benchmarks

- 7 with capture conditions

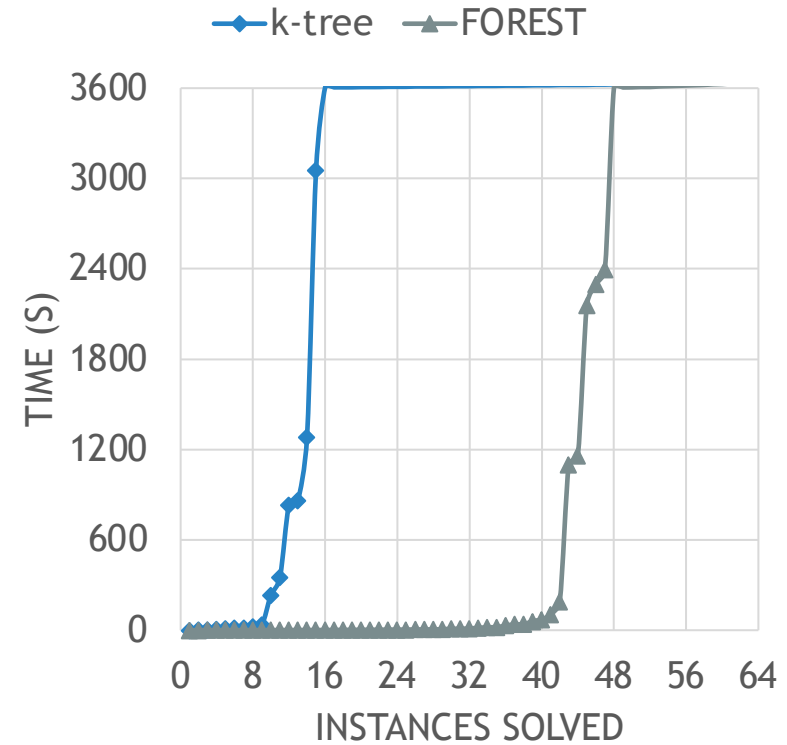
Real world:

- Existing validation libraries
- `regexlib.com`

## Results

# Representation

Timeout (s)	10	60	3600
<i>k</i> -tree	4	9	15
FOREST (Multi-tree)	31	39	47

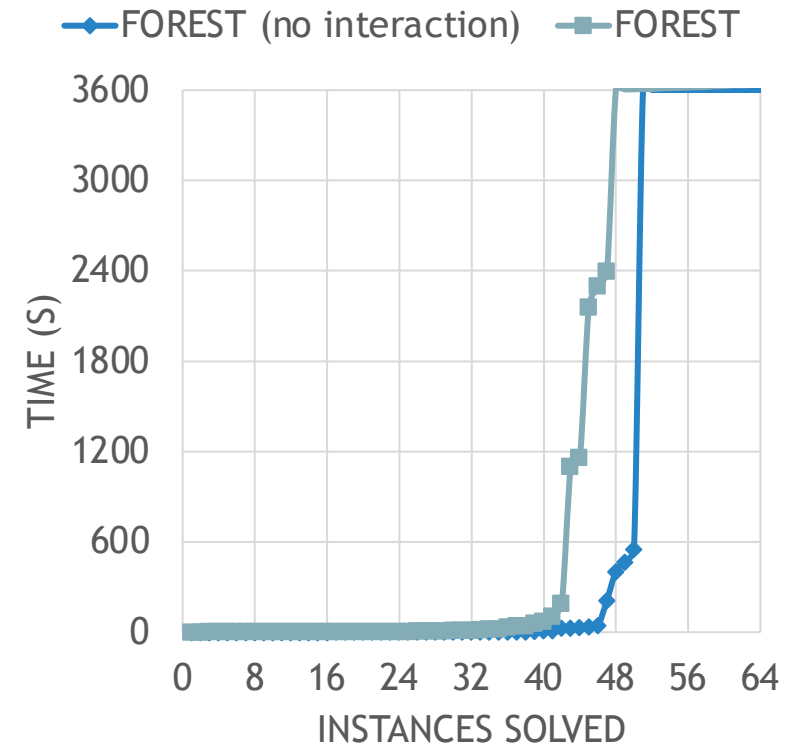


## Results

### Interaction

---

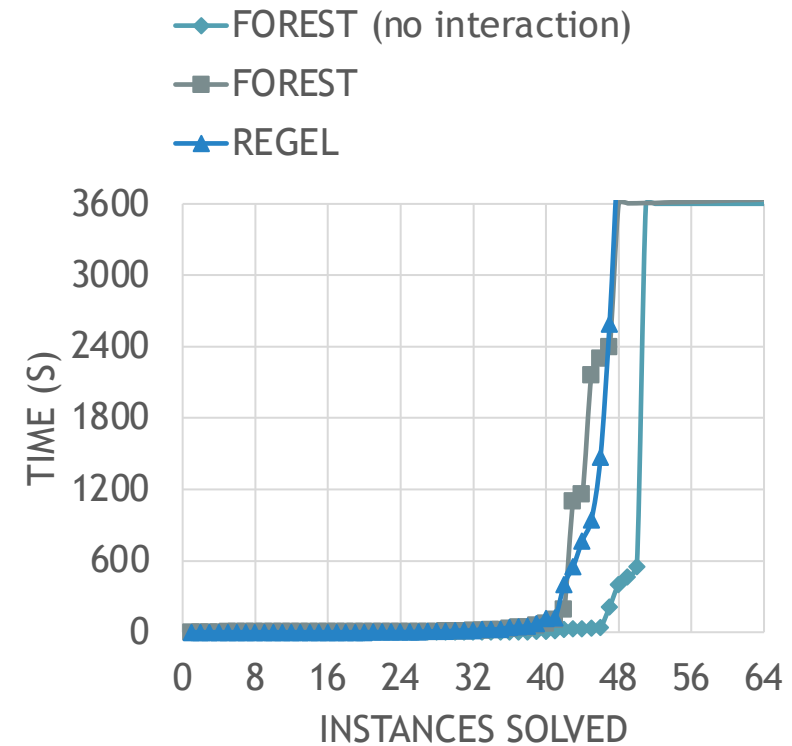
Timeout (s)	10	60	3600	Correct
FOREST (no interaction)	40	46	50	31
FOREST	31	39	47	45



## Results

### Comparison with REGEL

Timeout (s)	10	60	3600	Correct
FOREST (no interaction)	40	46	50	31
FOREST	31	39	47	45
REGEL	29	38	47	31





# Where to find FOREST ?

<https://github.com/marghrid/FOREST>

# Main contributions

---

- Multi-tree representation,
- Interaction model using distinguishing inputs for regexes,
- Capturing groups synthesis.



# FOREST: An Interactive Multi-tree Synthesizer for Regular Expressions

<https://github.com/marghrid/FOREST>