

Programming Languages and Compilers (CS 421)

Elsa L Gunter
2112 SC, UIUC



<https://courses.engr.illinois.edu/cs421/sp2023>

Based in part on slides by Mattox Beckman, as updated by Vikram Adve and Gul Agha



Midterm 2 Stats

Grade	Scale	Scale Class %	Bracket %	Promised curve \geq	Bracket
A	≥ 93	60.00%	60.00%	24%	24%
A-	≥ 90	66.27%	6.27%	35%	11%
B+	≥ 87	68.67%	2.41%	35%	0%
B	≥ 83	71.57%	2.89%	48%	13%
B-	≥ 80	73.49%	1.93%	66%	18%
C+	≥ 77	75.18%	1.69%	73%	7%
C	≥ 73	78.80%	3.61%	79%	6%
C-	≥ 70	81.20%	2.41%	89%	10%
D+	≥ 67	83.13%	1.93%	92%	3%
D	≥ 63	85.06%	1.93%	93%	1%
D-	≥ 60	86.51%	1.45%	96%	3%
F	< 60		13.49%		4%



Retake Possibility

- Sign up to retake at most one Midterm
- Won't open until Midterm 3 is finished
- Points = $(0.8 * \text{retake pts}) + (0.05 \text{ orig pts})$
- In CBTF last (full ?) week



Parser Code

- `<grammar>.mly` defines one parsing function per entry point
- Parsing function takes a lexing function (lexer buffer to token) and a lexer buffer as arguments
- Returns semantic attribute of corresponding entry point



Ocamlyacc Input

- File format:

%{

<header>

%}

<declarations>

%%

<rules>

%%

<trailer>



Ocamlyacc *<header>*

- Contains arbitrary Ocaml code
- Typically used to give types and functions needed for the semantic actions of rules and to give specialized error recovery
- May be omitted
- *<footer>* similar. Possibly used to call parser



Ocamlyacc <declarations>

- **%token** *symbol ... symbol*
- Declare given symbols as tokens
- **%token** <*type*> *symbol ... symbol*
- Declare given symbols as token constructors, taking an argument of type <*type*>
- **%start** *symbol ... symbol*
- Declare given symbols as entry points; functions of same names in <*grammar*>.ml



Ocamlyacc *<declarations>*

- **%type** *<type> symbol ... symbol*

Specify type of attributes for given symbols.

Mandatory for start symbols

- **%left** *symbol ... symbol*

- **%right** *symbol ... symbol*

- **%nonassoc** *symbol ... symbol*

Associate precedence and associativity to given symbols. Same line, same precedence; earlier line, lower precedence (broadest scope)



Ocamlyacc *<rules>*

- *nonterminal* :

symbol ... symbol { semantic_action }

| ...

| *symbol ... symbol { semantic_action }*

;

- Semantic actions are arbitrary Ocaml expressions
- Must be of same type as declared (or inferred) for *nonterminal*
- Access semantic attributes (values) of symbols by position: \$1 for first symbol, \$2 to second ...



Example - Base types

```
(* File: expr.ml *)
```

```
type expr =
```

```
  Term_as_Expr of term
```

```
  | Plus_Expr of (term * expr)
```

```
  | Minus_Expr of (term * expr)
```

```
and term =
```

```
  Factor_as_Term of factor
```

```
  | Mult_Term of (factor * term)
```

```
  | Div_Term of (factor * term)
```

```
and factor =
```

```
  Id_as_Factor of string
```

```
  | Parenthesized_Expr_as_Factor of expr
```



Example - Lexer (exprlex.ml)

```
{ (*open Exprparse*) }  
let numeric = ['0' - '9']  
let letter = ['a' - 'z' 'A' - 'Z']  
rule token = parse  
| "+" {Plus_token}  
| "-" {Minus_token}  
| "*" {Times_token}  
| "/" {Divide_token}  
| "(" {Left_parenthesis}  
| ")" {Right_parenthesis}  
| letter (letter|numeric|"_"|"_")* as id {Id_token id}  
| [' ' '\t' '\n'] {token lexbuf}  
| eof {EOL}
```



Example - Parser (exprparse.mly)

```
%{ open Expr
```

```
%}
```

```
%token <string> Id_token
```

```
%token Left_parenthesis Right_parenthesis
```

```
%token Times_token Divide_token
```

```
%token Plus_token Minus_token
```

```
%token EOL
```

```
%start main
```

```
%type <expr> main
```

```
%%
```



Example - Parser (exprparse.mly)

expr:

term

{ Term_as_Expr \$1 }

| term Plus_token expr

{ Plus_Expr (\$1, \$3) }

| term Minus_token expr

{ Minus_Expr (\$1, \$3) }



Example - Parser (exprparse.mly)

term:

factor

{ Factor_as_Term \$1 }

| factor Times_token term

{ Mult_Term (\$1, \$3) }

| factor Divide_token term

{ Div_Term (\$1, \$3) }



Example - Parser (exprparse.mly)

factor:

Id_token

{ Id_as_Factor \$1 }

| Left_parenthesis expr Right_parenthesis

{ Parenthesized_Expr_as_Factor \$2 }

main:

| expr EOL

{ \$1 }



Example - Using Parser

```
# #use "expr.ml";;
```

```
...
```

```
# #use "exprparse.ml";;
```

```
...
```

```
# #use "exprlex.ml";;
```

```
...
```

```
# let test s =
```

```
  let lexbuf = Lexing.from_string (s^"\n") in  
    main token lexbuf;;
```




Example - Using Parser

```
# test "a + b";;
```

```
- : expr =
```

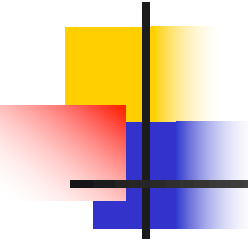
```
Plus_Expr
```

```
(Factor_as_Term (Id_as_Factor "a"),  
Term_as_Expr (Factor_as_Term  
(Id_as_Factor "b")))
```



LR Parsing

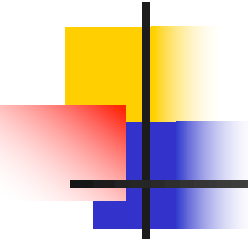
- Read tokens left to right (L)
- Create a rightmost derivation (R)
- How is this possible?
- Start at the bottom (left) and work your way up
- Last step has only one non-terminal to be replaced so is right-most
- Working backwards, replace mixed strings by non-terminals
- Always proceed so that there are no non-terminals to the right of the string to be replaced



Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$= \bullet (0 + 1) + 0$ shift



Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

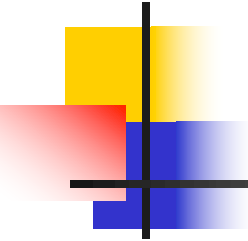
$\langle \text{Sum} \rangle \Rightarrow$

$$= (\bullet 0 + 1) + 0$$

$$= \bullet (0 + 1) + 0$$

shift

shift



Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$$\Rightarrow (0 \bullet + 1) + 0$$

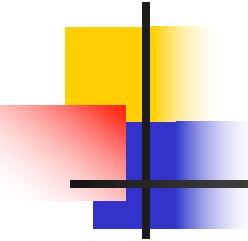
$$= (\bullet 0 + 1) + 0$$

$$= \bullet (0 + 1) + 0$$

reduce

shift

shift



Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$$= (\langle \text{Sum} \rangle \bullet + 1) + 0$$

shift

$$\Rightarrow (0 \bullet + 1) + 0$$

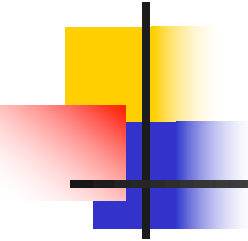
reduce

$$= (\bullet 0 + 1) + 0$$

shift

$$= \bullet (0 + 1) + 0$$

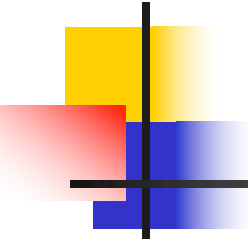
shift



Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$=$	$(\langle \text{Sum} \rangle + \bullet 1) + 0$	shift
$=$	$(\langle \text{Sum} \rangle \bullet + 1) + 0$	shift
\Rightarrow	$(0 \bullet + 1) + 0$	reduce
$=$	$(\bullet 0 + 1) + 0$	shift
$=$	$\bullet (0 + 1) + 0$	shift



Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$	reduce
$= (\langle \text{Sum} \rangle + \bullet 1) + 0$	shift
$= (\langle \text{Sum} \rangle \bullet + 1) + 0$	shift
$\Rightarrow (0 \bullet + 1) + 0$	reduce
$= (\bullet 0 + 1) + 0$	shift
$= \bullet (0 + 1) + 0$	shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle \bullet) + 0$ reduce
 $\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$ reduce
 $= (\langle \text{Sum} \rangle + \bullet 1) + 0$ shift
 $= (\langle \text{Sum} \rangle \bullet + 1) + 0$ shift
 $\Rightarrow (0 \bullet + 1) + 0$ reduce
 $= (\bullet 0 + 1) + 0$ shift
 $= \bullet (0 + 1) + 0$ shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$= (\langle \text{Sum} \rangle \bullet) + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle \bullet) + 0$ reduce
 $\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$ reduce
 $= (\langle \text{Sum} \rangle + \bullet 1) + 0$ shift
 $= (\langle \text{Sum} \rangle \bullet + 1) + 0$ shift
 $\Rightarrow (0 \bullet + 1) + 0$ reduce
 $= (\bullet 0 + 1) + 0$ shift
 $= \bullet (0 + 1) + 0$ shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$\Rightarrow (\langle \text{Sum} \rangle) \bullet + 0$ reduce
 $= (\langle \text{Sum} \rangle \bullet) + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle \bullet) + 0$ reduce
 $\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$ reduce
 $= (\langle \text{Sum} \rangle + \bullet 1) + 0$ shift
 $= (\langle \text{Sum} \rangle \bullet + 1) + 0$ shift
 $\Rightarrow (0 \bullet + 1) + 0$ reduce
 $= (\bullet 0 + 1) + 0$ shift
 $= \bullet (0 + 1) + 0$ shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$= \langle \text{Sum} \rangle \bullet + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle) \bullet + 0$ reduce
 $= (\langle \text{Sum} \rangle \bullet) + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle \bullet) + 0$ reduce
 $\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$ reduce
 $= (\langle \text{Sum} \rangle + \bullet 1) + 0$ shift
 $= (\langle \text{Sum} \rangle \bullet + 1) + 0$ shift
 $\Rightarrow (0 \bullet + 1) + 0$ reduce
 $= (\bullet 0 + 1) + 0$ shift
 $= \bullet (0 + 1) + 0$ shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$

$= \langle \text{Sum} \rangle + \bullet 0$ shift
 $= \langle \text{Sum} \rangle \bullet + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle) \bullet + 0$ reduce
 $= (\langle \text{Sum} \rangle \bullet) + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle \bullet) + 0$ reduce
 $\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$ reduce
 $= (\langle \text{Sum} \rangle + \bullet 1) + 0$ shift
 $= (\langle \text{Sum} \rangle \bullet + 1) + 0$ shift
 $\Rightarrow (0 \bullet + 1) + 0$ reduce
 $= (\bullet 0 + 1) + 0$ shift
 $= \bullet (0 + 1) + 0$ shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle \Rightarrow$
 $\Rightarrow \langle \text{Sum} \rangle + 0 \bullet$ reduce
 $= \langle \text{Sum} \rangle + \bullet 0$ shift
 $= \langle \text{Sum} \rangle \bullet + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle) \bullet + 0$ reduce
 $= (\langle \text{Sum} \rangle \bullet) + 0$ shift
 $\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle \bullet) + 0$ reduce
 $\Rightarrow (\langle \text{Sum} \rangle + 1 \bullet) + 0$ reduce
 $= (\langle \text{Sum} \rangle + \bullet 1) + 0$ shift
 $= (\langle \text{Sum} \rangle \bullet + 1) + 0$ shift
 $\Rightarrow (0 \bullet + 1) + 0$ reduce
 $= (\bullet 0 + 1) + 0$ shift
 $= \bullet (0 + 1) + 0$ shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle$	$\Rightarrow \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$	●	reduce
	$\Rightarrow \langle \text{Sum} \rangle + 0$	●	reduce
	$= \langle \text{Sum} \rangle +$	● 0	shift
	$= \langle \text{Sum} \rangle$	● + 0	shift
	$\Rightarrow (\langle \text{Sum} \rangle)$	● + 0	reduce
	$= (\langle \text{Sum} \rangle$	●) + 0	shift
	$\Rightarrow (\langle \text{Sum} \rangle + \langle \text{Sum} \rangle$	●) + 0	reduce
	$\Rightarrow (\langle \text{Sum} \rangle + 1$	●) + 0	reduce
	$= (\langle \text{Sum} \rangle +$	● 1) + 0	shift
	$= (\langle \text{Sum} \rangle$	● + 1) + 0	shift
	$\Rightarrow (0$	● + 1) + 0	reduce
	$= ($	● 0 + 1) + 0	shift
	$=$	● (0 + 1) + 0	shift

Example: $\langle \text{Sum} \rangle = 0 \mid 1 \mid (\langle \text{Sum} \rangle)$
 $\mid \langle \text{Sum} \rangle + \langle \text{Sum} \rangle$

$\langle \text{Sum} \rangle$	●	\Rightarrow	$\langle \text{Sum} \rangle + \langle \text{Sum} \rangle$	●	reduce	
		\Rightarrow	$\langle \text{Sum} \rangle + 0$	●	reduce	
		=	$\langle \text{Sum} \rangle +$	●	0	shift
		=	$\langle \text{Sum} \rangle$	●	+ 0	shift
		\Rightarrow	$(\langle \text{Sum} \rangle)$	●	+ 0	reduce
		=	$(\langle \text{Sum} \rangle$	●) + 0	shift
		\Rightarrow	$(\langle \text{Sum} \rangle + \langle \text{Sum} \rangle$	●) + 0	reduce
		\Rightarrow	$(\langle \text{Sum} \rangle + 1$	●) + 0	reduce
		=	$(\langle \text{Sum} \rangle +$	●	1) + 0	shift
		=	$(\langle \text{Sum} \rangle$	●	+ 1) + 0	shift
		\Rightarrow	$(0$	●	+ 1) + 0	reduce
		=	$($	●	0 + 1) + 0	shift
		=	●	$(0 + 1) + 0$	shift	



Example

$$(0 + 1) + 0$$





Example

$$(\quad 0 \quad + \quad 1 \quad) \quad + \quad 0$$



4/4/23



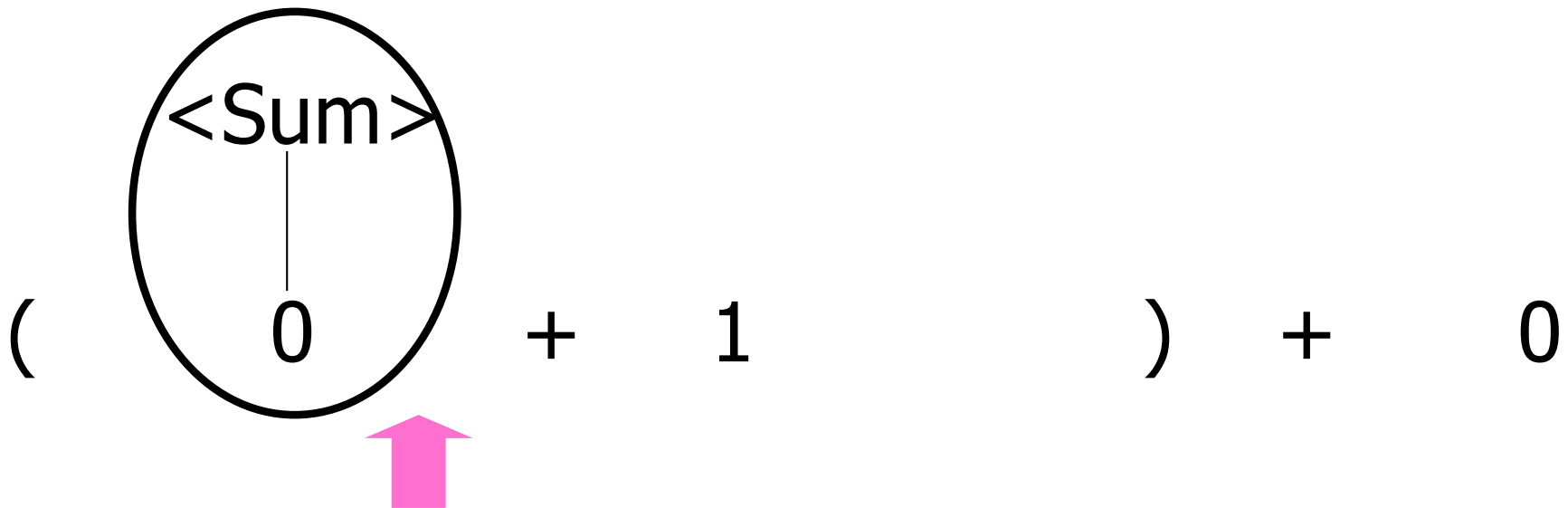
Example

(0 + 1) + 0



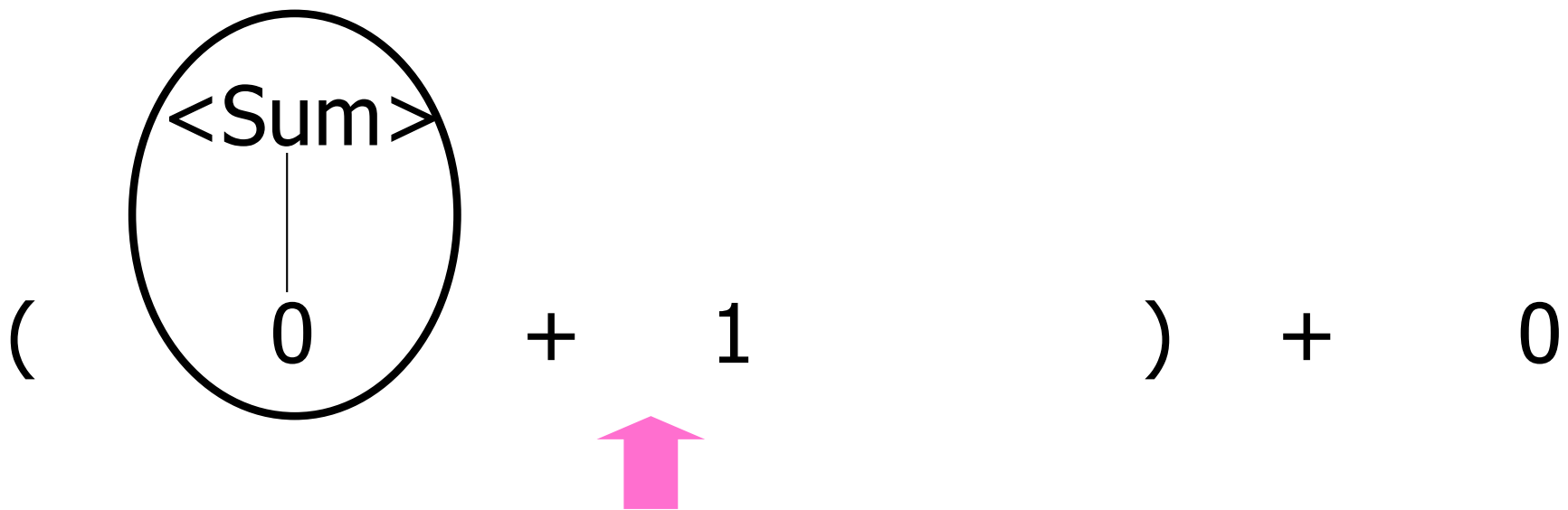


Example





Example



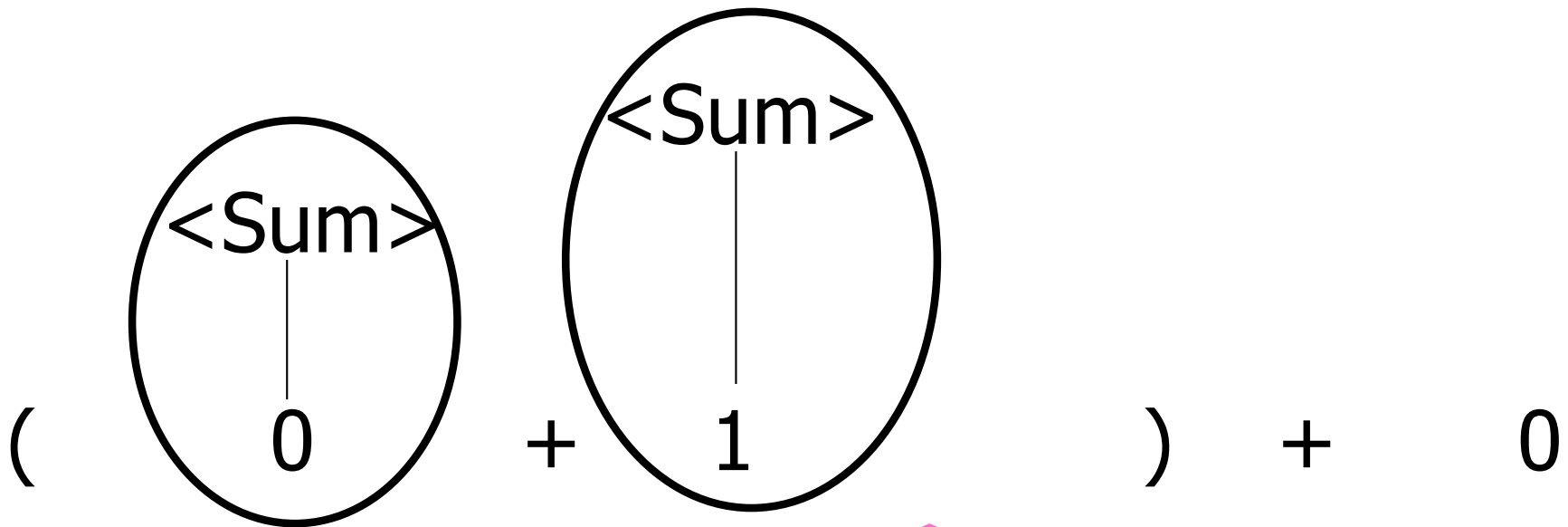


Example



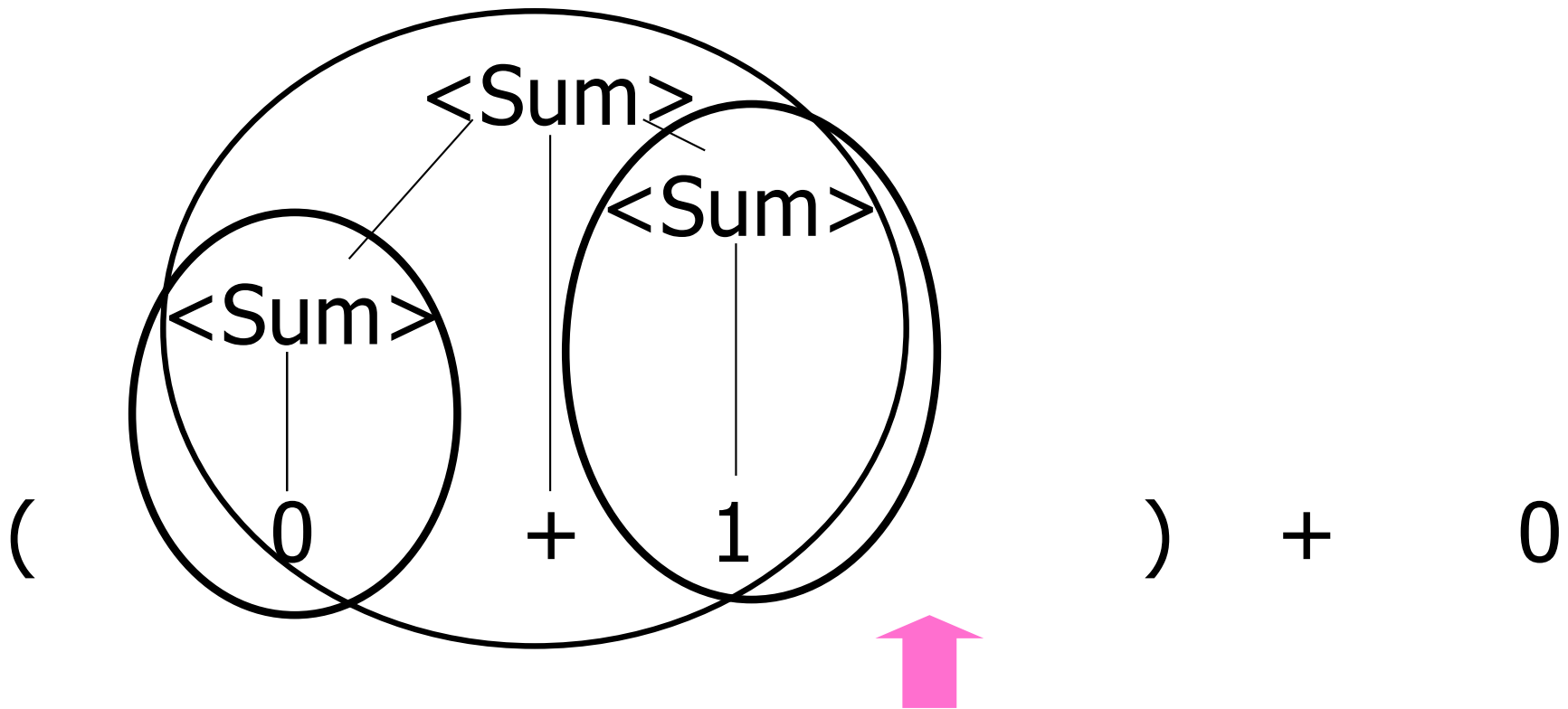


Example



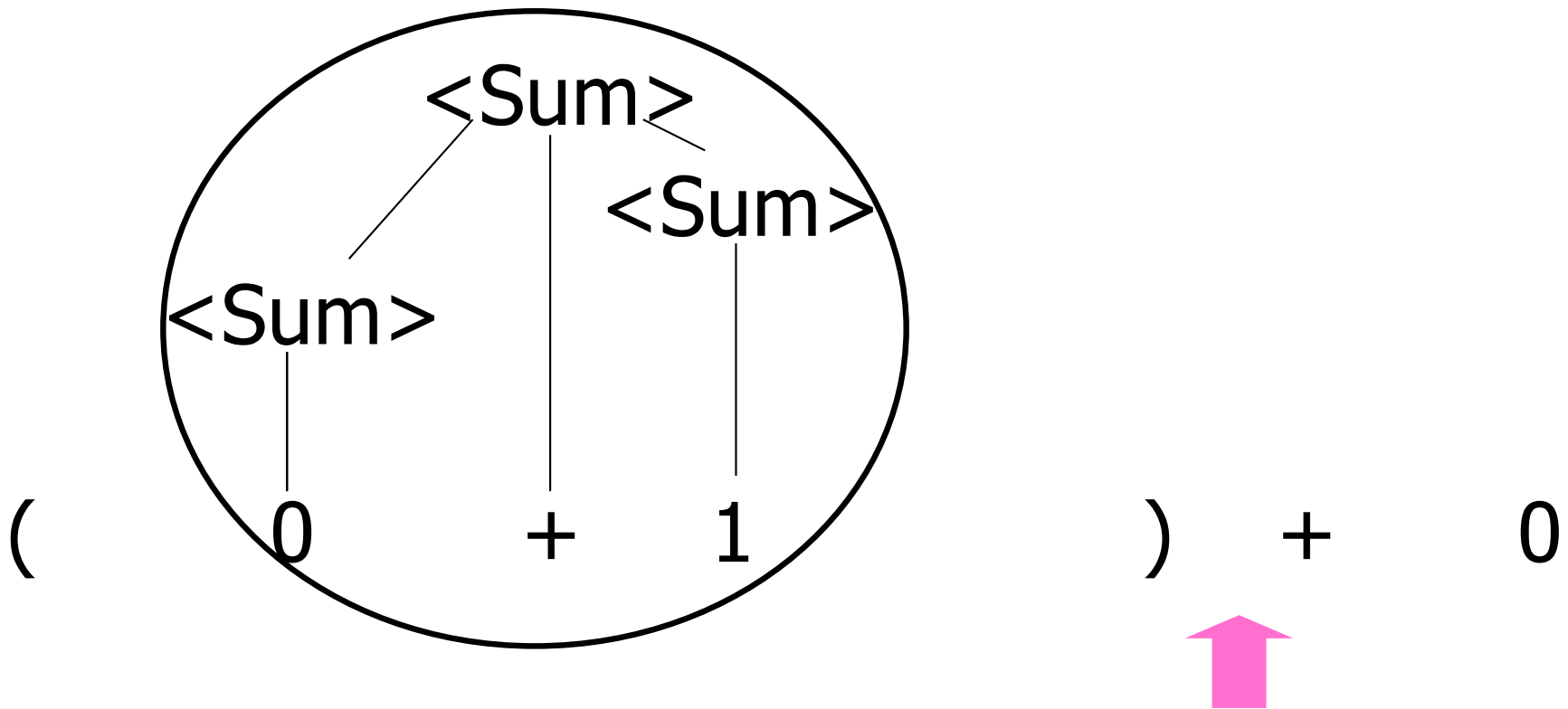


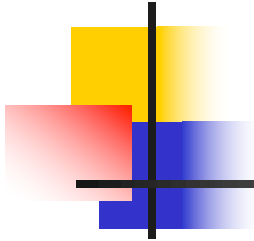
Example



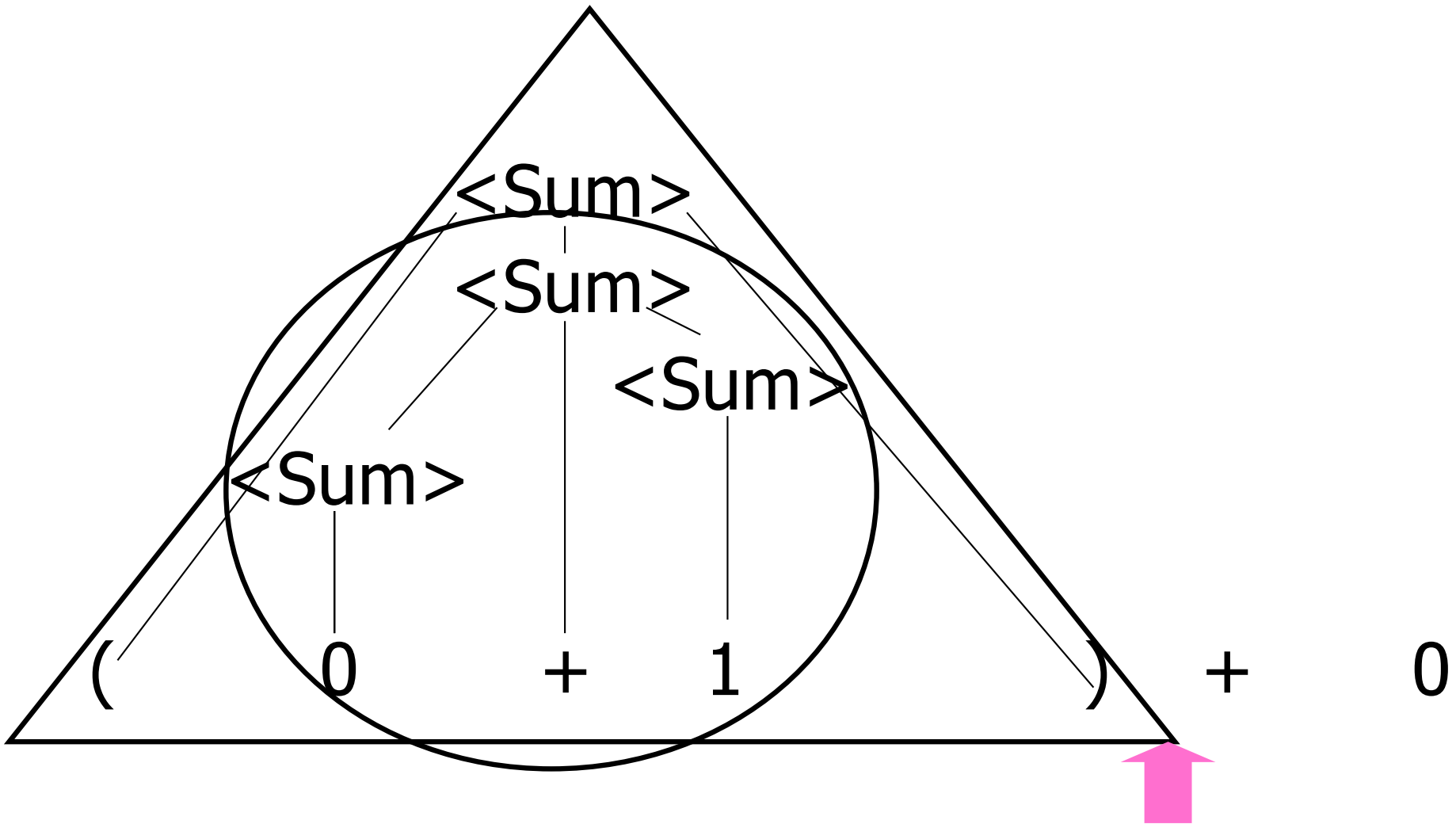


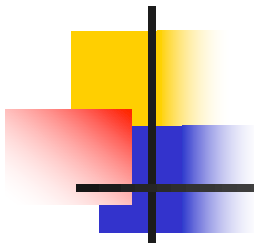
Example



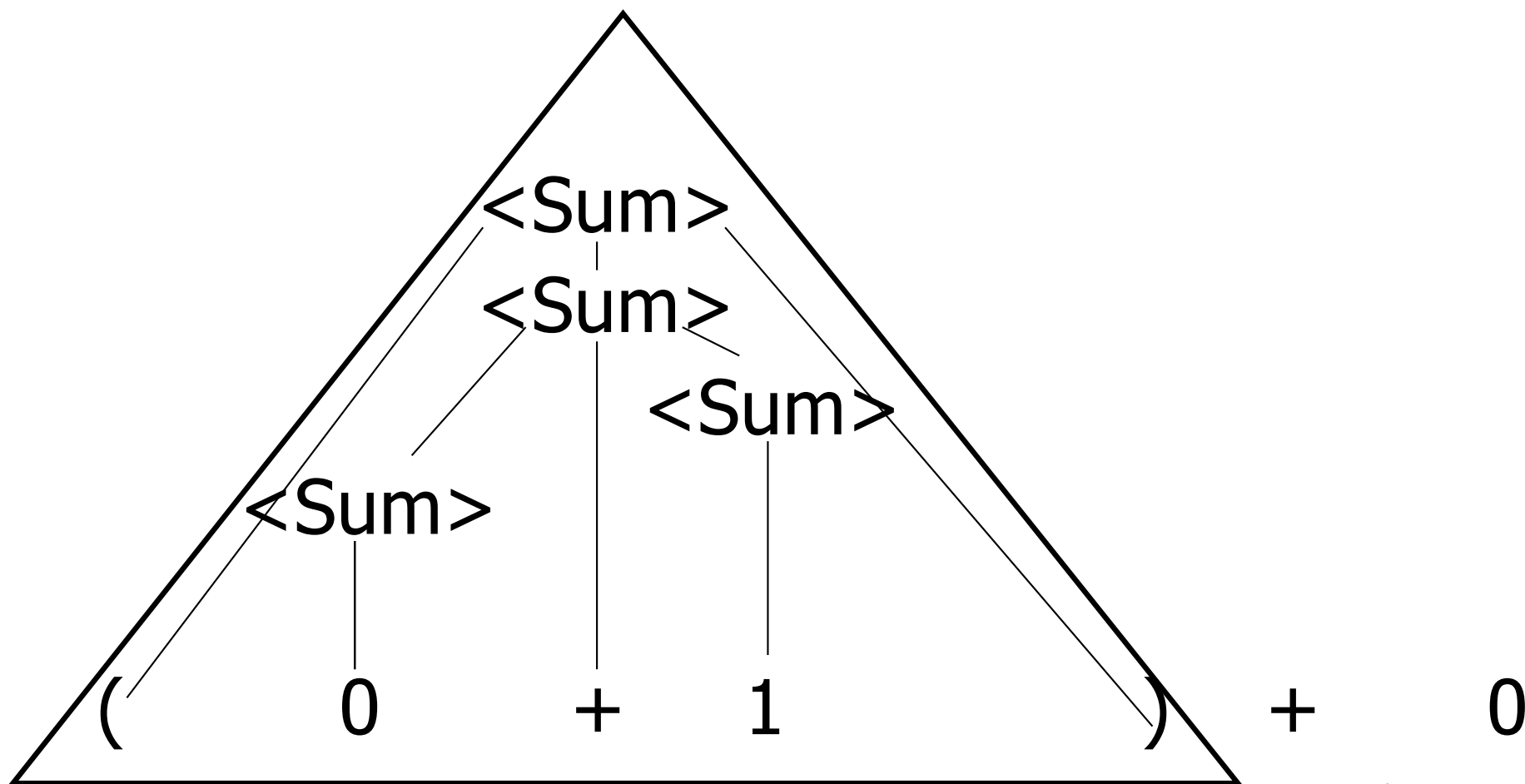


Example



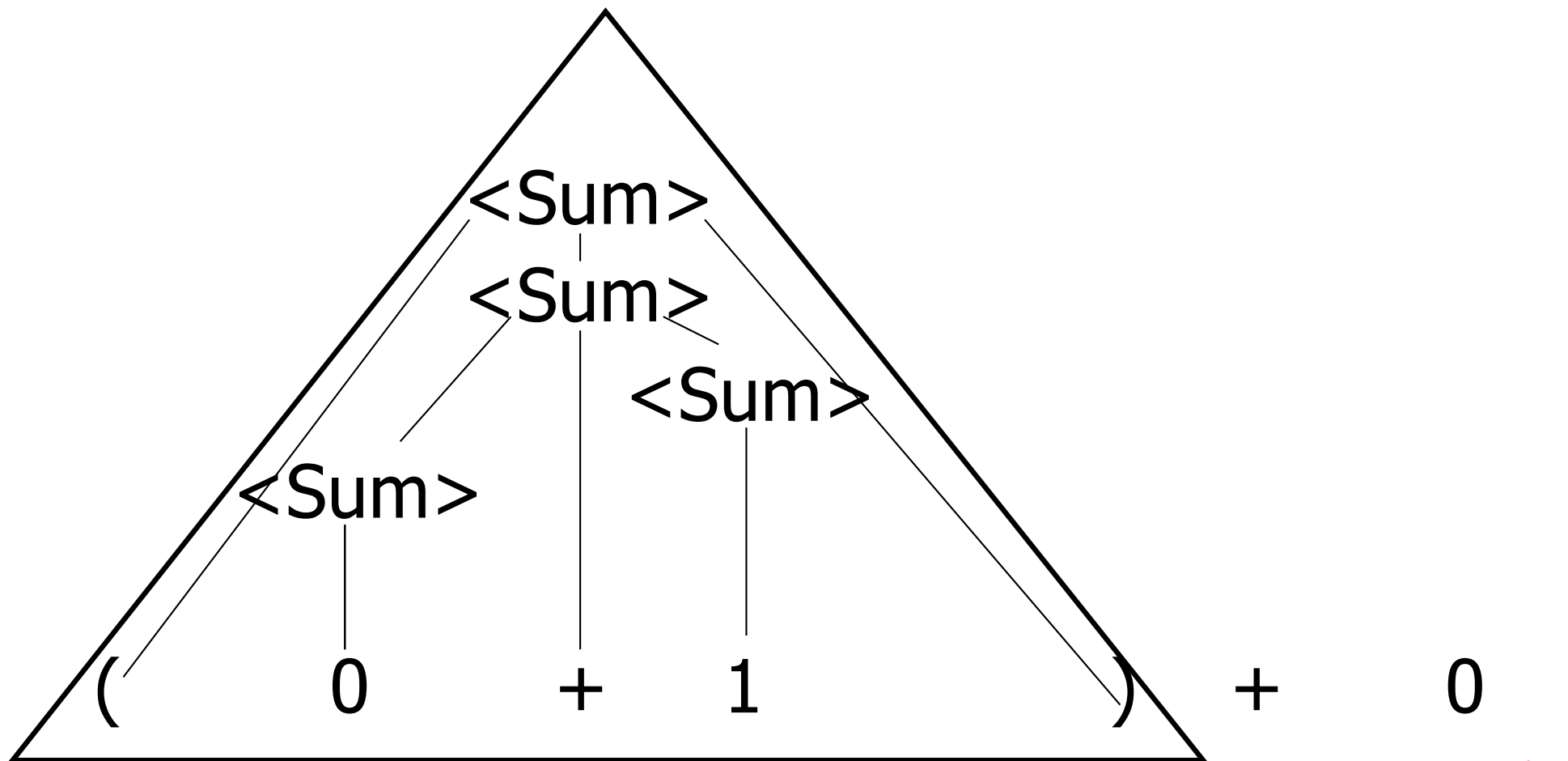


Example



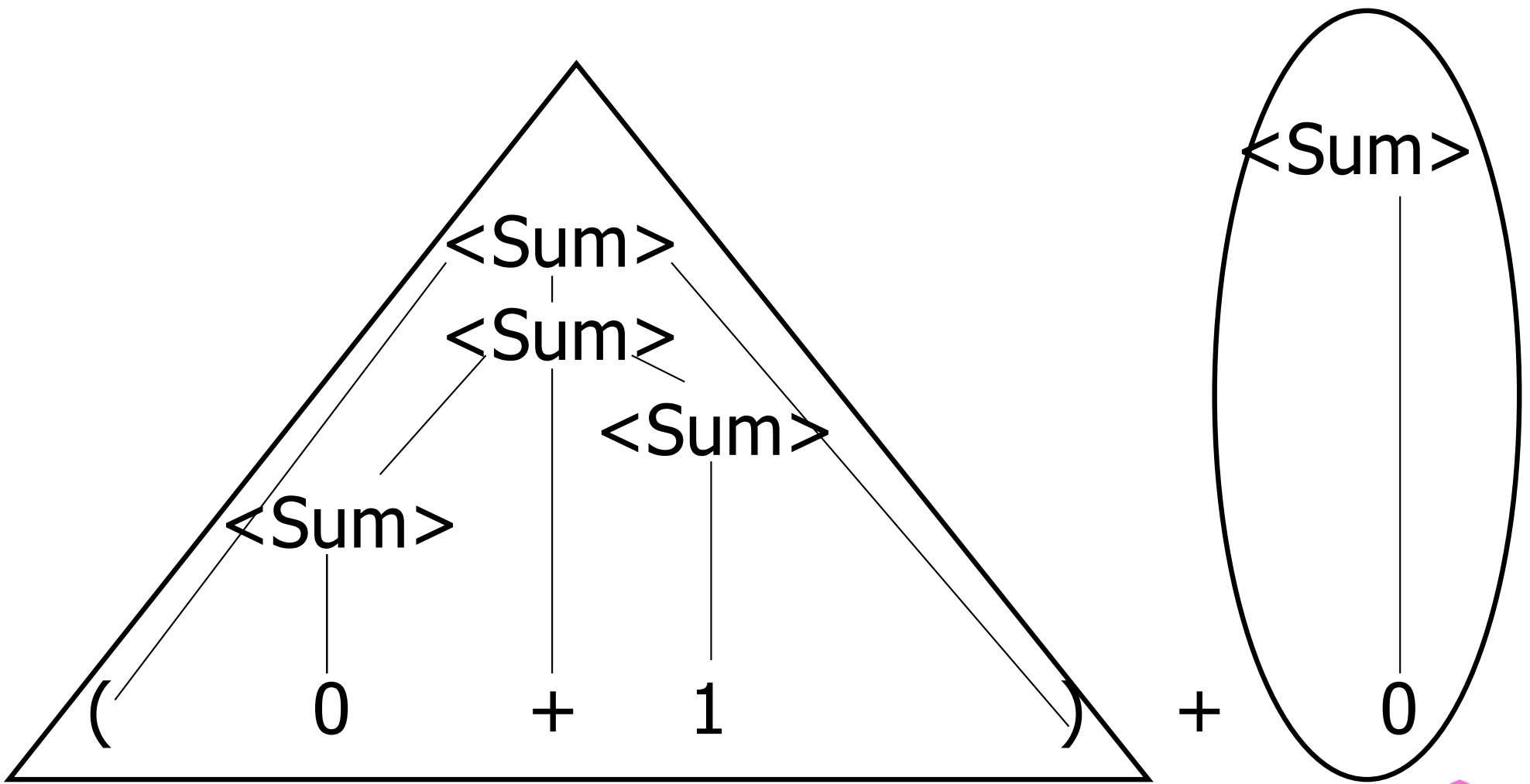


Example



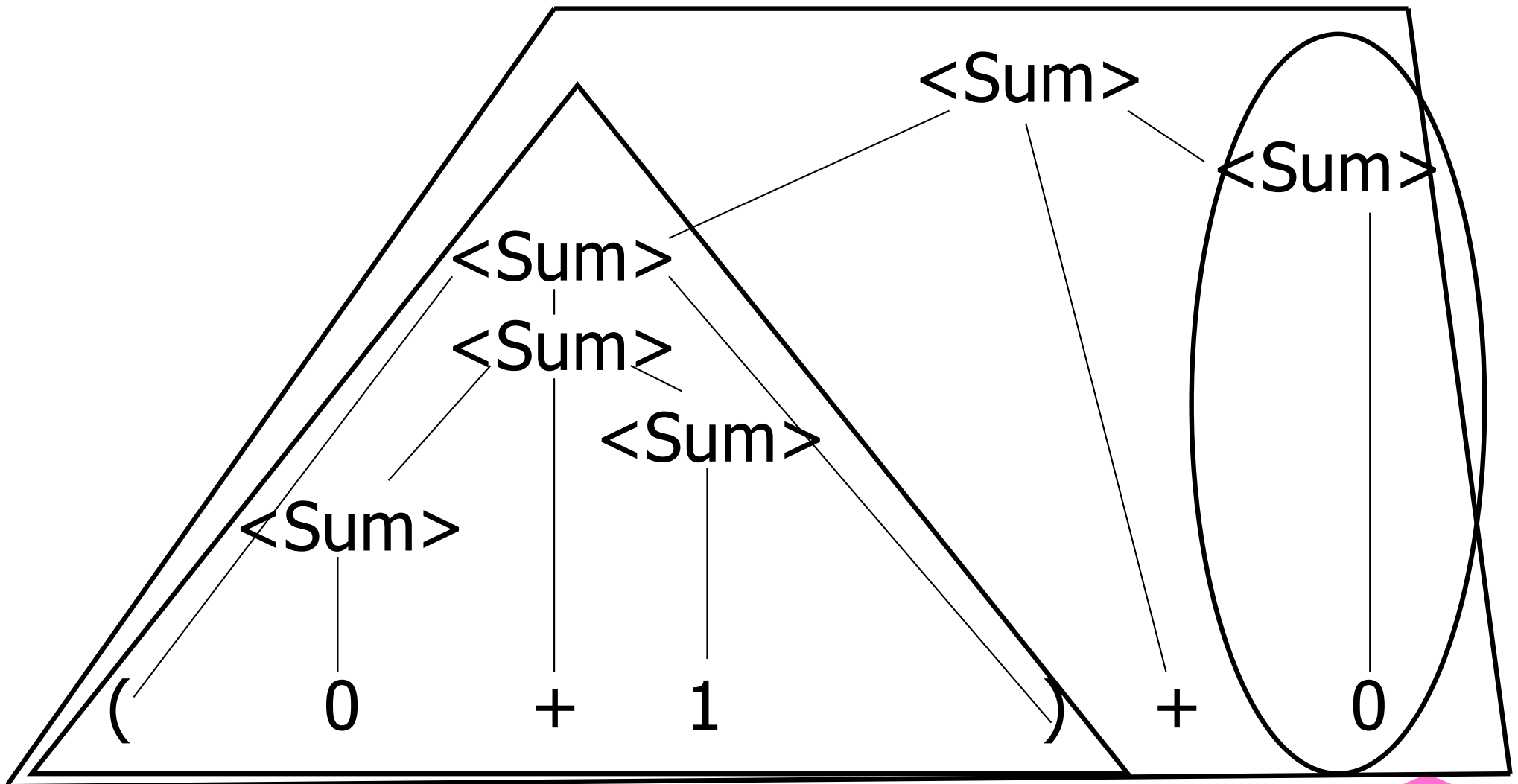


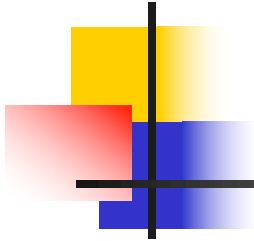
Example





Example





Example

