Introduction to Parsing Using
Lex and Yacc

Uday Khedker

(www.cse.iitb.ac.in/˜uday)

Department of Computer Science and Engineering,
Indian Institute of Technology, Bombay

June 2024

ACM Summer School
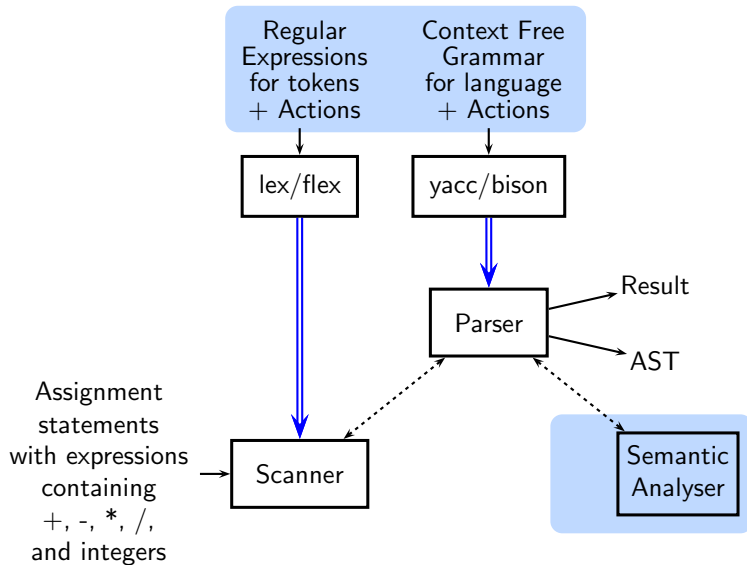on Compilers for
AI/ML

Topic:

Lex & Yacc

Section:

Introduction to Lex

ACM Summer School
on Compilers for
AI/ML

Topic:

Lex & Yacc

Section:

- First lex script `minimal.l`

- Add the "." rule to `minimal.l`

- Use `encode-vovels.l`. Take the input `vowel-input.txt`

- Use `decode-vovels.l`. Pipe the out of encoding to decoding and have fun ;-)

- Use `num.l`

- Show `kw.l`, then `kw-id-incorrect-*.l`, and then `kw-id-correct.l`

- Show `discarding-input.l`

Introduction to Yacc

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- `exp1.y` merely shows reduction of a number to an expression
- `exp2.y` shows the construction of plus expression
    - On input `1 + 2 + 3 + 4 + 5`, it first reads all the numbers and then starts constructing the expression
- `exp3.y` fixes the above problem by making $+$ left associative
- `exp4.y` includes $+$ and `*`
    - On input `1 + 2 * 3` it seems fine
    - On input `1 * 2 + 3` it does not seem fine
- `exp5.y` fixes the above problem
- `unambiguous-exp.yr` shows how an unambiguous grammar can be used to avoid using precedences and associativities

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

The Interaction Between Scanner and Parser

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- Terminals and non-terminals get defined by the grammar

- Scanner identifies the tokens and communicates the details to the parser

Token Name	Token Lexemes	Token Value	Token Code
Number	"10"	10	NUM
	"345"	345	
	"03"	3	
+ operator	"+"		'+'
* operator	"*"		'*'

The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
AI/ML

Topic:

Lex & Yacc
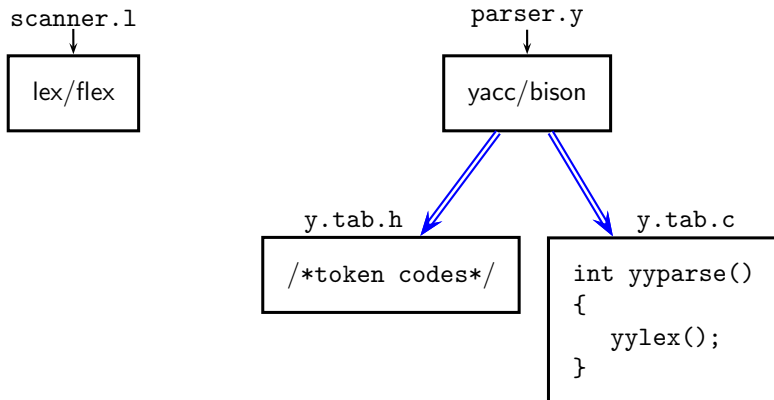
Section:

```
scanner.l
```
↓

| lex/flex |

```
parser.y
```
↓

| yacc/bison |

The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

```
scanner.l
```
↓
┌─────────┐
│ lex/flex │
└─────────┘

```
parser.y
```
↓
┌─────────────┐
│ yacc/bison │
└─────────────┘

```
y.tab.h
```
┌──────────────────┐
│ /*token codes*/ │
└──────────────────┘

```
y.tab.c
```
┌──────────────────┐
│ int yyparse()   │
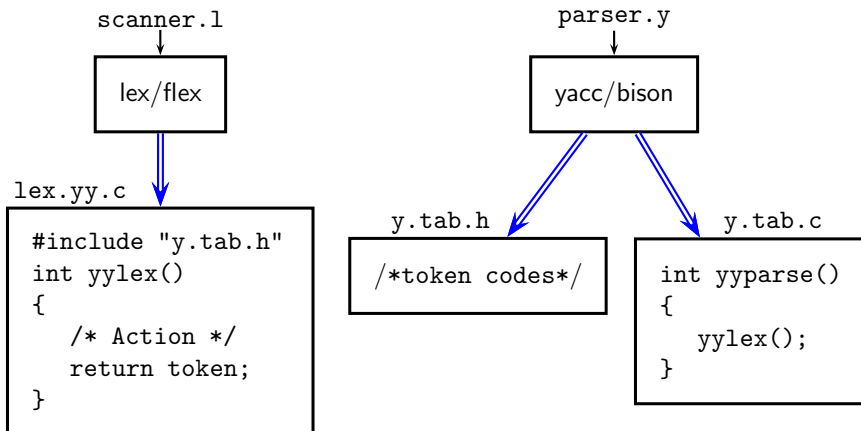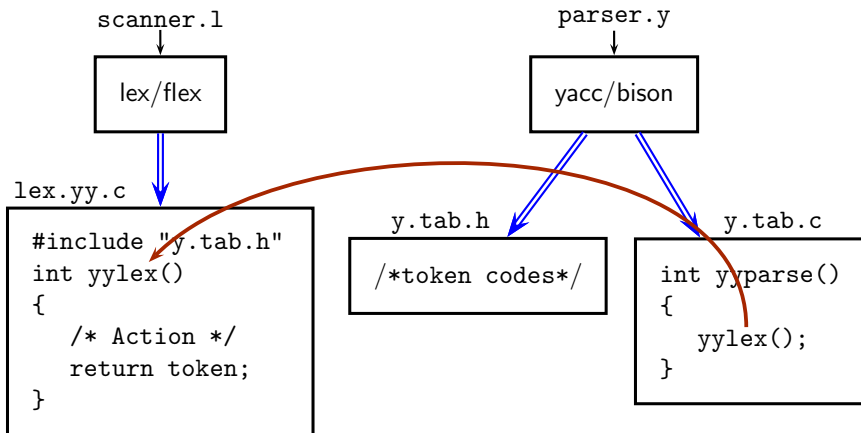│ {   │
│   yylex();   │
│ }   │
└──────────────────┘

The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

scanner.l

lex/flex

parser.y

yacc/bison

lex.yy.c

```
#include "y.tab.h"
int yylex()
{
    /* Action */
    return token;
}
```

y.tab.h

```
/*token codes*/
```

y.tab.c

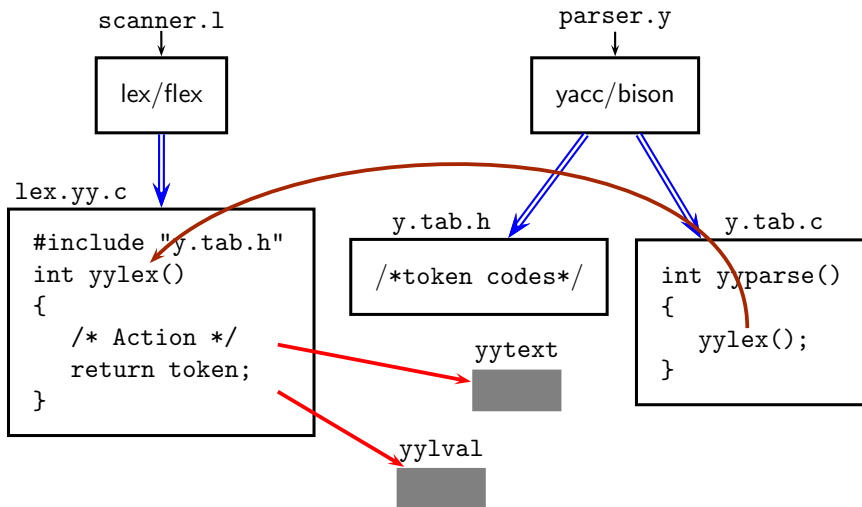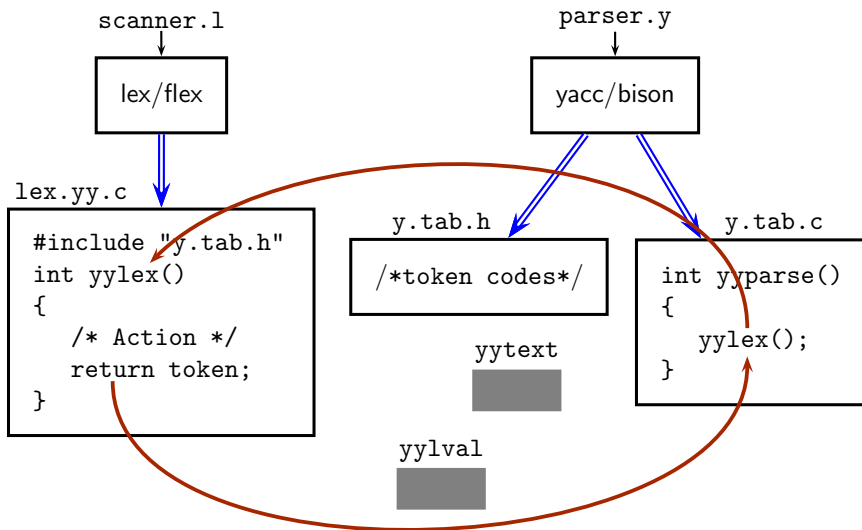```
int yyparse()
{
    yylex();
}
```

# The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

# The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
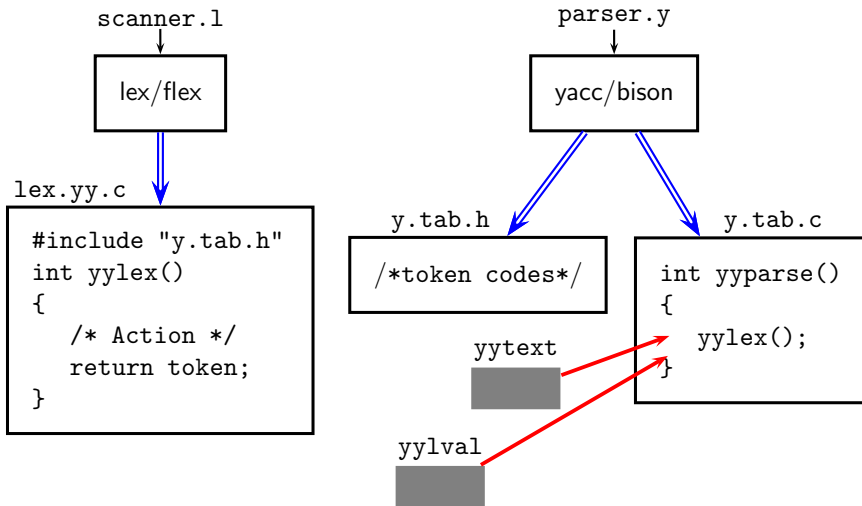AI/ML

Topic:
Lex & Yacc
Section:

# The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
AI/ML

Topic:

Lex & Yacc

Section:

scanner.l

lex/flex

parser.y

yacc/bison

lex.yy.c

```
#include "y.tab.h"
int yylex()
{
    /* Action */
    return token;
}
```

y.tab.h

```
/*token codes*/
```

y.tab.c

```
int yyparse()
{
    yylex();
}
```

yytext

yylval

The Interaction Between Scanner and Parser

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

scanner.l

```
lex/flex
```

parser.y

```
yacc/bison
```

lex.yy.c

```
#include "y.tab.h"
int yylex()
{
    /* Action */
    return token;
}
```

y.tab.h

```
/*token codes*/
```

y.tab.c

```
int yyparse()
{
    yylex();
}
```

yytext

yylval

An Overview of Shift Reduce Parsing

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- The process of parsing

<div style="text-align:center">

10   +   20   *   3 | eof |
NUM

</div>

Parsing
Stack

Action: Shift NUM

An Overview of Shift Reduce Parsing

ACM Summer School
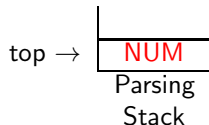on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- The process of parsing

$$10 \quad + \quad 20 \quad * \quad 3 \quad \boxed{eof}$$
$$+$$

top $\rightarrow$
NUM

Parsing
Stack

Action: Reduce by "Expr: NUM"

An Overview of Shift Reduce Parsing

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- The process of parsing

$$10 \quad + \quad 20 \quad * \quad 3 \quad \boxed{\text{eof}}$$
$$+$$

top $\rightarrow$ | Expr |

Parsing
Stack

Action: Shift $+$

An Overview of Shift Reduce Parsing

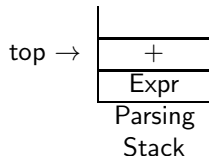ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- The process of parsing

<pre>
    10   +   20   *   3 | eof |
                NUM
</pre>

top →

+
Expr

Parsing
Stack

Action: Shift NUM

An Overview of Shift Reduce Parsing

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:
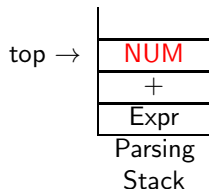
- Grammar

```
Expr : Expr '+' Expr
     | Expr '*' Expr
     | NUM
```

- The process of parsing

Action: Reduce by "Expr: NUM"

An Overview of Shift Reduce Parsing

ACM Summer School
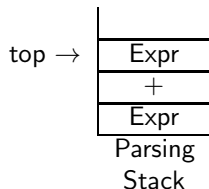on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- The process of parsing

<div>
10   +   20   *   3 | eof |
</div>

top →

Expr
+
Expr

Parsing
Stack

Action: Shift ∗ or Reduce by "Expr: Expr + Expr"?

An Overview of Shift Reduce Parsing

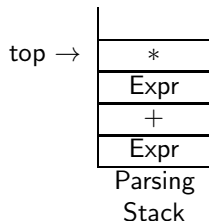ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr : Expr '+' Expr
     | Expr '*' Expr
     | NUM
```

- The process of parsing

$$10 \quad + \quad 20 \quad * \quad \underset{\text{NUM}}{3} \quad \boxed{\text{eof}}$$

top $\rightarrow$

$*$
Expr
$+$
Expr

Parsing
Stack

Action: Shift NUM

An Overview of Shift Reduce Parsing

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```
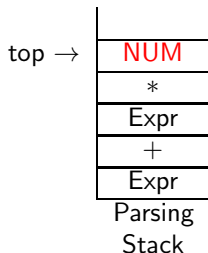
- The process of parsing

$$10 \quad + \quad 20 \quad * \quad 3 \quad \boxed{\text{eof}}$$
$$\text{EOF}$$

top $\rightarrow$

NUM
$*$
Expr
$+$
Expr

Parsing
Stack

Action: Reduce by "Expr: NUM"

An Overview of Shift Reduce Parsing

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar
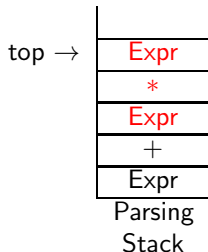
```
Expr : Expr '+' Expr
     | Expr '*' Expr
     | NUM
```

- The process of parsing

$$10 \quad + \quad 20 \quad * \quad 3 \quad \boxed{\text{eof}}$$
$$\text{EOF}$$

top $\rightarrow$

Expr
*
Expr
+
Expr

Parsing
Stack

Action: Reduce by "Expr: Expr * Expr"

An Overview of Shift Reduce Parsing

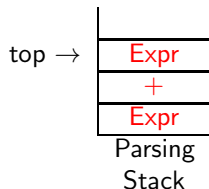ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr : Expr '+' Expr
     | Expr '*' Expr
     | NUM
```

- The process of parsing

$$10 \quad + \quad 20 \quad * \quad 3 \quad \boxed{eof}$$
$$\text{EOF}$$

top $\rightarrow$

Expr
+
Expr

Parsing
Stack

Action: Reduce by "Expr: Expr + Expr"

An Overview of Shift Reduce Parsing

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar

```
Expr  :  Expr '+' Expr
      |  Expr '*' Expr
      |  NUM
```

- The process of parsing

$$10 \quad + \quad 20 \quad * \quad 3 \quad \boxed{\text{eof}}$$
$$\text{EOF}$$

top $\rightarrow$

Expr

Parsing
Stack

Action: Accept

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
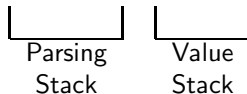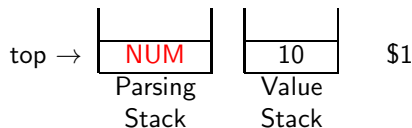Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
      |  NUM
```

- The process of parsing and attribute evaluation

$$10 \quad + \quad 20 \quad \boxed{eof}$$

NUM

Parsing Stack   Value Stack

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
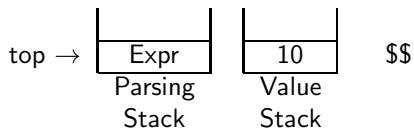Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
         |  NUM
```

- The process of parsing and attribute evaluation

```
                         10     +     20     eof
                                +
```

```
         top →  | NUM  |   |  10  |    $1
                Parsing     Value
                 Stack      Stack
```

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
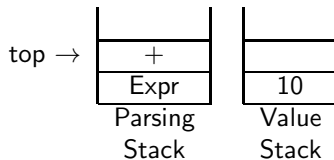Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
      |  NUM
```

- The process of parsing and attribute evaluation

$$10 \quad + \quad 20 \quad \boxed{eof}$$
$$+$$

top $\rightarrow$ | Expr | | 10 | $$
Parsing Stack   Value Stack

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
      |  NUM
```

- The process of parsing and attribute evaluation

```
10     +     20     eof
             NUM
```

```
top →  |       |     |       |
       |   +   |     |       |
       | Expr  |     |  10   |
       Parsing      Value
       Stack        Stack
```

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
         |  NUM
```

- The process of parsing and attribute evaluation

An Overview of Attribute Evaluation

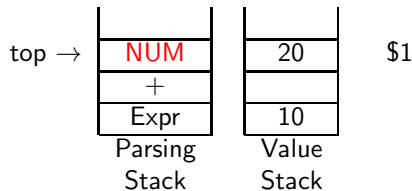ACM Summer School
on Compilers for
AI/ML

Topic:

Lex & Yacc

Section:

- Grammar and the associated actions
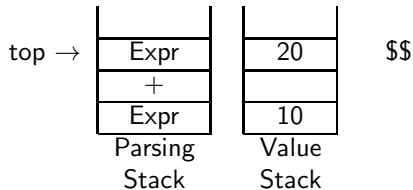
```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
         |  NUM
```

- The process of parsing and attribute evaluation

$$10 \quad + \quad 20 \quad \boxed{\text{eof}}$$
EOF

top →	Expr	20	$$
	+		
	Expr	10	
	Parsing Stack	Value Stack	

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
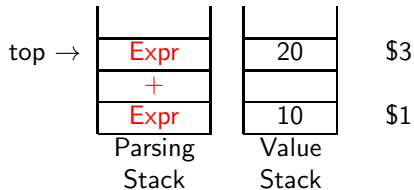AI/ML

Topic:

Lex & Yacc

Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
         |  NUM
```

- The process of parsing and attribute evaluation

An Overview of Attribute Evaluation

ACM Summer School
on Compilers for
AI/ML

Topic:
Lex & Yacc
Section:

- Grammar and the associated actions

```
Expr  :  Expr '+' Expr    { $$ = $1 + $3;}
         |  NUM
```

- The process of parsing and attribute evaluation