

## Plan for This Week

---

### Context Free Grammars

- model for specifying programming languages
- why not just use regular expressions?
- example grammar
- derivations

### Parse trees

### Syntax-directed translation

- using syntax-directed translation to interpret MiniSVG

### Top-down Predictive Parsing

## MiniSVG Grammar

---

svg -> SVG\_START elem\_list SVG\_END

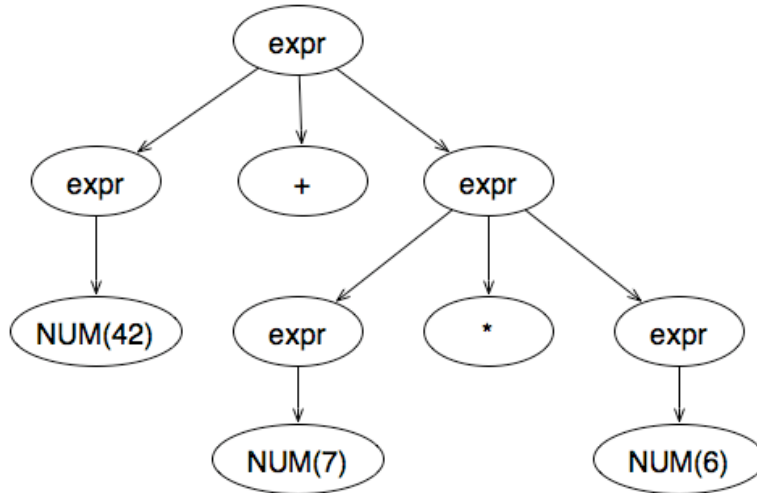
elem\_list -> elem elem\_list | epsilon

elem -> RECT\_START KW\_X EQ NUM KW\_Y EQ NUM KW\_WIDTH  
EQ NUM KW\_HEIGHT EQ NUM KW\_FILL EQ COLOR ELEM\_END

| CIRCLE\_START KW\_CX EQ NUM KW\_CY EQ NUM KW\_R EQ NUM  
KW\_FILL EQ COLOR ELEM\_END

| LINE\_START KW\_X1 EQ NUM KW\_Y1 EQ NUM KW\_X2 EQ NUM  
KW\_Y2 EQ NUM KW\_STROKE EQ COLOR ELEM\_END

## Parse Tree Example



## Another Parse Tree Exam

### Grammar

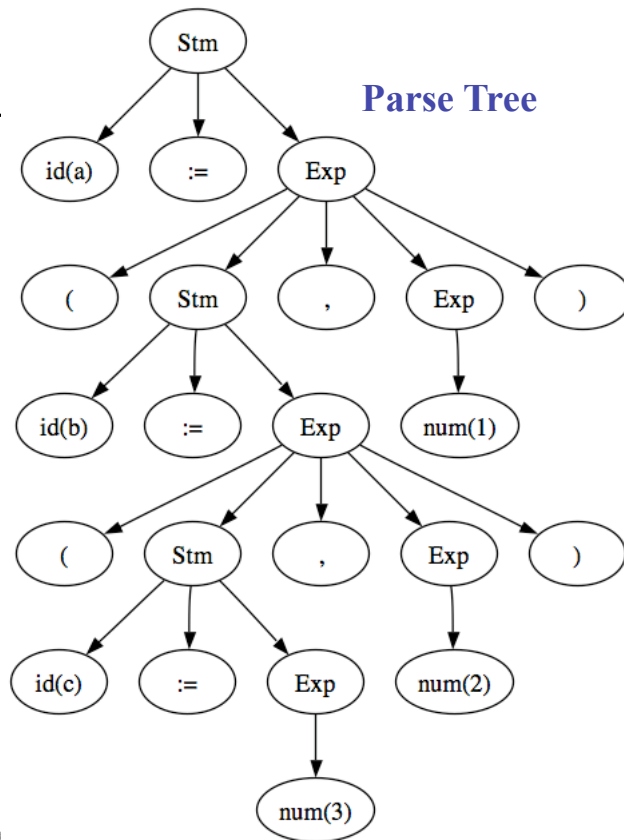
Stm  $\rightarrow$  id := Exp

Exp  $\rightarrow$  num

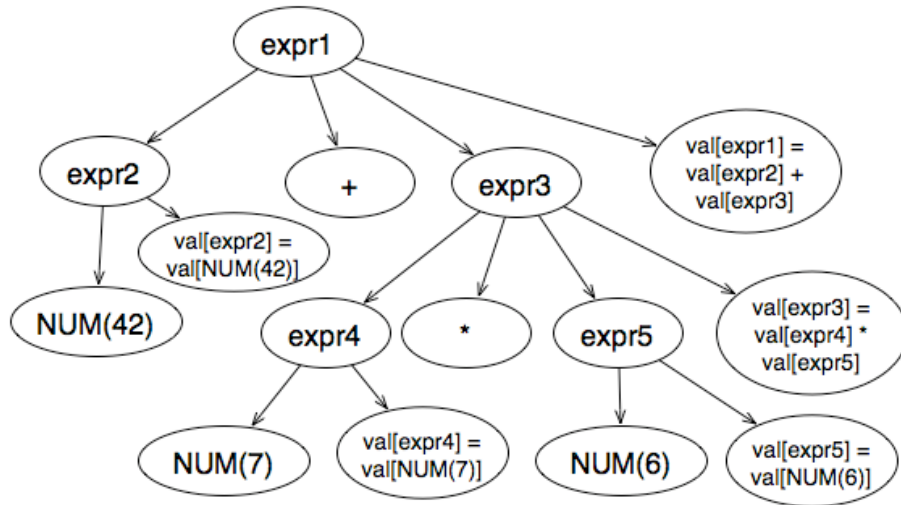
Exp  $\rightarrow$  ( Stm, Exp )

### String

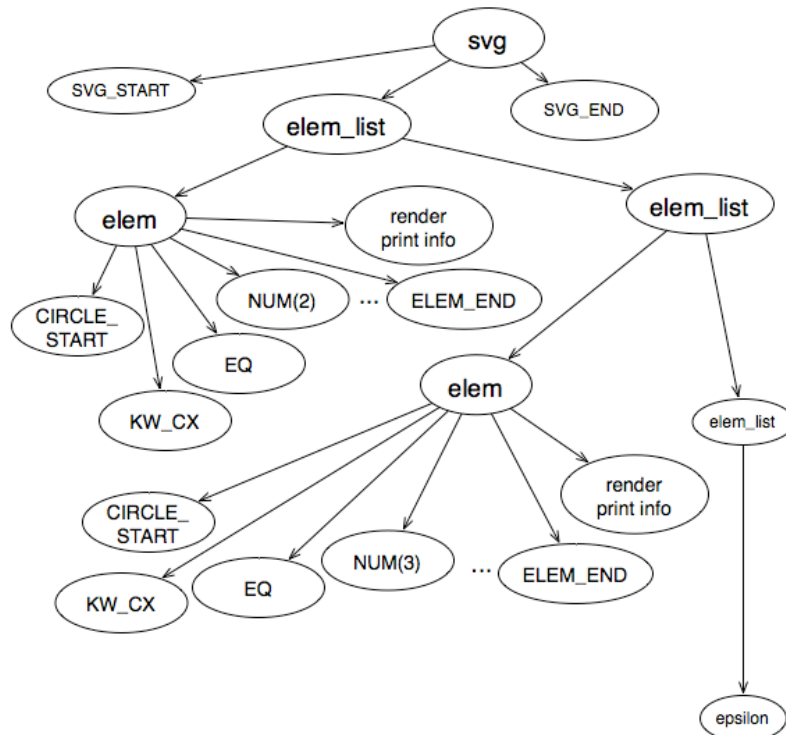
a := ( b := ( c := 3, 2 ), 1 )



## Semantic Rules for Expression Example



## Example Parse Tree for MiniSVG



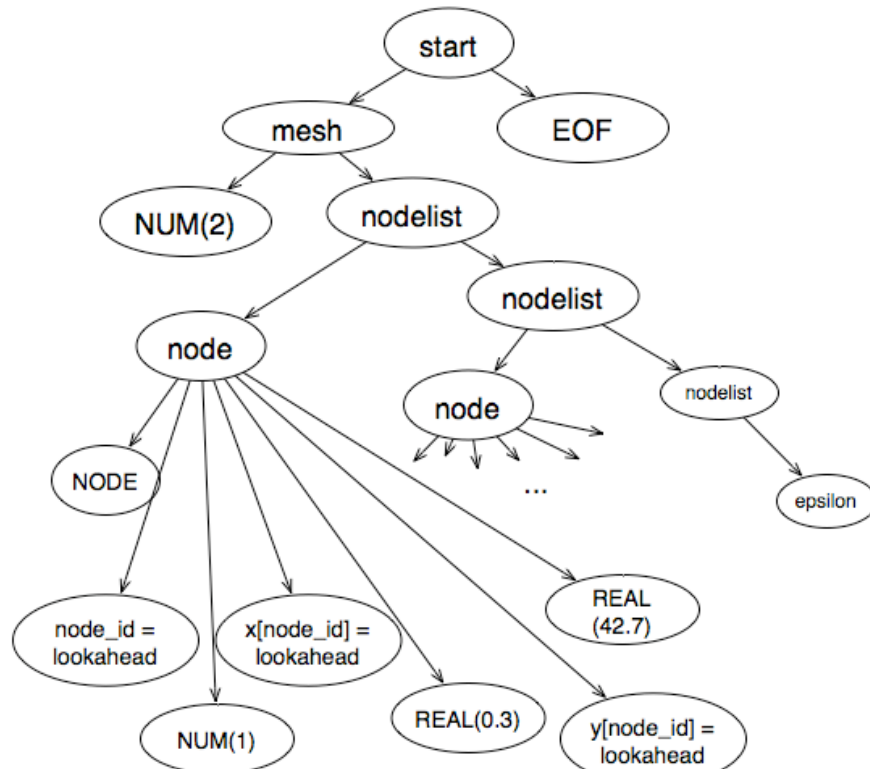
## Example Predictive Parser

- (1) start  $\rightarrow$  mesh EOF
- (2) mesh  $\rightarrow$  NUM nodelist NUM elemelist
- (3a & b) nodelist  $\rightarrow$   $\epsilon$  | node nodelist
- (4) node  $\rightarrow$  NODE NUM REAL REAL // node\_id, x, y
- (5a & b) elemelist  $\rightarrow$   $\epsilon$  | elem elemelist
- (6a) elem  $\rightarrow$  TRI NUM NUM NUM NUM // elem\_id, 3 node ids
- (6b) elem  $\rightarrow$  SQR NUM NUM NUM NUM NUM //elem\_id,4 node ids

```

void start() { switch(m_lookahead) {
    case NUM: mesh(); match(Token.Tag.EOF); break;
    default:  throw new ParseException(...);
}}
void mesh() { switch(this.m_lookahead) {
    case NUM: num_nodes = ((Num)m_lookahead).value; match(NUM);
              nodelist();
              num_elem = ((Num)m_lookahead).value; match(NUM);
              elemelist(); break;
    default:  throw new ParseException(...);
}}
void nodelist() { switch(m_lookahead) {
    case NUM: break; // nodelist -> epsilon
    case NODE: node(); nodelist(); break; // nodelist -> node nodelist
    default:  throw new ParseException(...);
}}

```



## MiniSVG Grammar

---

svg -> SVG\_START elem\_list SVG\_END

elem\_list -> elem elem\_list | epsilon

elem -> RECT\_START KW\_X EQ NUM KW\_Y EQ NUM KW\_WIDTH  
EQ NUM KW\_HEIGHT EQ NUM KW\_FILL EQ COLOR ELEM\_END

| CIRCLE\_START KW\_CX EQ NUM KW\_CY EQ NUM KW\_R EQ NUM  
KW\_FILL EQ COLOR ELEM\_END

| LINE\_START KW\_X1 EQ NUM KW\_Y1 EQ NUM KW\_X2 EQ NUM  
KW\_Y2 EQ NUM KW\_STROKE EQ COLOR ELEM\_END