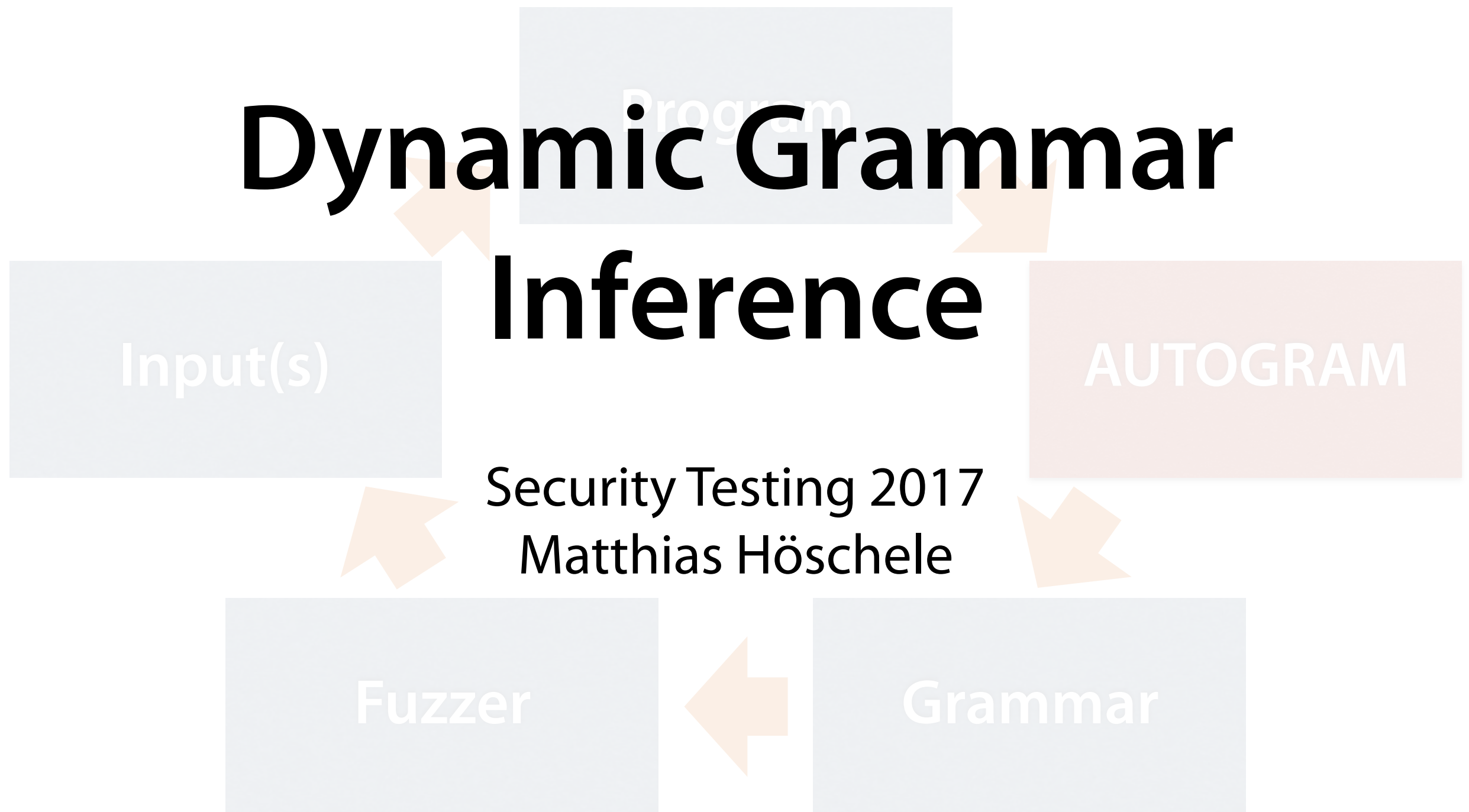


Dynamic Grammar Inference



Security Testing 2017
Matthias Hörschele



UNIVERSITÄT
DES
SAARLANDES

CISPA
Center for IT-Security, Privacy
and Accountability

<https://www.st.cs.uni-saarland.de/models/autogram/>

Fuzz Testing

[;x1-GPZ+wcckc];,N9J+?#6^6\e?]9lu2_%'4GX"0VUB[E/r
~fApu6b8<{%siq8Zh.6{V,hr?;{Ti.r3PlxMMMv6{xS^+'Hq!
AxB"YXRS@!Kd6;wtAMefFWM(`IJ_<1~o}z3K(CCzRH JllvHz>_*.
\\>JrIU32~eGP?IR=bF3+;y\$3lodQ<B89!5"W2fK*vE7v{')KC-
i,c{<[~m!]o;{.'}Gj\X}EtYetrbpY@aGZ1{P!AZU7x#4(Rtn!
q4nCwqol^y6}0IKo=*JK~;zMKV=9Nai:wxu{J&UV#HaU)*BiC<),`
+t*gka<W=Z.%T5WGHZpl30D<Pq>&]BS6R&j?#tP7iaV}-}\?
[_[Z^LBMPG-FKj'xwuZ1=Q`^5,\$N\$Q@[!CuRzJ2DivBy!
^zkhdf3C5PAkR?V hnl3='i2Qx]D
\$qs4O`1@fevnG'2\11Vf3piU37@55ap\zlyl'"f,
\$ee,J4Gw:cgNKLie3nx9(`efSlg6#[K"@WjhZ}r[Scun&sBCS,T[/
vY'pduwgzDIVNy7'rnzxNwl)(ynBa>%lb`;`9fG]P_0hdG~\$@6
3]KAeEnQ7IU)3Pn,0)G/6N-wyzj/MTd#A;r



Fuzz Testing

[;x1-GPZ+wcckc];,N9J+?#6^6\e?]9lu2_%'4GX"0VUB[E/r
~fApu6b8<{%siq8Zh.6{V,hr?;{Ti.r3PlxMMMv6{xS^+'Hq!
AxB"YXRS@!Kd6;wtAMefFWM(`IJ_<1~o}z3K(CCzRH JllvHz>_*.
\>JrIU32~eGP?IR=bF3+;y\$3lodQ<B89!5"W2fK*vE7v{'})KC-
i,c{<[~m!]o;{.'}Gj\X}EtYetrbpY@aGZ1{P!AZU7x#4(Rtn!
q4nCwqol^y6}0IKo=*JK~;zMKV=9Nai:wxu{J&UV#HaU)*BiC<),`
+t*gka<W=Z.%T5WGHZpl30D<Pq>&]BS6R&j?#tP7iaV}-}\?
[_[Z^LBMPG-FKj'xwuZ1=Q`^5,\$N\$Q@[!CuRzJ2DivBy!
^zkhdf3C5PAkR?V hnl3='i2Qx]D
\$qs4O`1@fevnG'2\11Vf3piU37@55ap\zlyl"f,
\$ee,J4Gw:cgNKLie3nx9(`efSlg6#[K"@WjhZ}r[Scun&sBCS,T[/
vY'pduwgzDIVNy7'rnzxNwl)(ynBa>%lb`;`9fG]P_0hdG~\$@6
3]KAeEnQ7IU)3Pn,0)G/6N-wyzj/MTd#A;r



Syntax Error

It's not very
effective...

An Input Grammar

If Statement

IfStatement^{full} ⇒

if ParenthesizedExpression Statement^{full}

| **if** ParenthesizedExpression Statement^{noShortIf} **else** Statement^{full}

IfStatement^{noShortIf} ⇒ **if** ParenthesizedExpression Statement^{noShortIf} **else** Statement^{noShortIf}

Switch Statement

SwitchStatement ⇒

switch ParenthesizedExpression { }

| **switch** ParenthesizedExpression { CaseGroups LastCaseGroup }

CaseGroups ⇒

«empty»

| CaseGroups CaseGroup

CaseGroup ⇒ CaseGuards BlockStatementsPrefix

LastCaseGroup ⇒ CaseGuards BlockStatements

CaseGuards ⇒

CaseGuard

| CaseGuards CaseGuard

CaseGuard ⇒

Grammar-Based Fuzzing

```
var haystack = "foo";  
var re_text = "^foo";  
haystack = "foo";  
re_text = "foo";  
var re = new  
re.test(haystack);  
RegExp.input = Number();  
print(RegExp.$1);
```



30 Chromium + Mozilla Security Rewards
53,000 US\$ in Bug Bounties

Holler, Herzig, Zeller "Fuzzing with Code Fragments", USENIX 2012

Context Free Grammars

$$G = (V, \Sigma, R, S)$$

V Non-Terminal Symbols

Σ Terminal Symbols

R Productions, Relation V to $(V \cup \Sigma)^*$

S Start Symbol from V

Creating Grammars



Learning Grammars

<http://user:pass@www.google.com:80/path>



Program

Learning Grammars

`http://user:pass@www.google.com:80/path`

`http` – protocol

Learning Grammars

`http://user:pass@www.google.com:80/path`

`http` – protocol

`www.google.com` – host name

Learning Grammars

`http://user:pass@www.google.com:80/path`

<code>http</code>	– protocol
<code>www.google.com</code>	– host name
<code>80</code>	– port

Learning Grammars

`http://user:pass@www.google.com:80/path`

- `http` – protocol
- `www.google.com` – host name
- `80` – port
- `user pass` – login

Learning Grammars

`http://user:pass@www.google.com:80/path`

<code>http</code>	– protocol
<code>www.google.com</code>	– host name
<code>80</code>	– port
<code>user pass</code>	– login
<code>path</code>	– page request

Learning Grammars

`http://user:pass@www.google.com:80/path`

<code>http</code>	– protocol
<code>www.google.com</code>	– host name
<code>80</code>	– port
<code>user pass</code>	– login
<code>path</code>	– page request
<code>:// @ : /</code>	– terminals

Learning Grammars

http://user:pass@www.google.com:80/path

http

– protocol

www.google.com

– host name

80

– port

user pass

– login

path

– page request

:// : @ : /

– terminals

processed in
*different
functions*

stored in
*different
variables*

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)

|
param: protocol

|
param: host

|
param: port

|
param: authority

|
param: userinfo

|
param: path

|
param: query

|
param: ref

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

`java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)`

|
param: protocol

| `http`

param: host

|
param: port

|
param: authority

|
param: userinfo

|
param: path

|
param: query

|
param: ref

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)

param: protocol

| http

param: host

| www.google.com

param: port

param: authority

param: userinfo

param: path

param: query

param: ref

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)

param: protocol

| http

param: host

| www.google.com

param: port

param: authority

param: userinfo

| user:password

param: path

param: query

param: ref

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)

param: protocol

| http

param: host

| www.google.com

param: port

| 80

param: authority

param: userinfo

| user:password

param: path

param: query

param: ref

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)

param: protocol

| http

param: host

| www.google.com

param: port

| 80

param: authority

param: userinfo

| user:password

param: path

| /command

param: query

param: ref

http ://user:password@www.google.com:80 /command?foo=bar&lorem=ipsum#fragment

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)

param: protocol

| http

param: host

| www.google.com

param: port

| 80

param: authority

param: userinfo

| user:password

param: path

| /command

param: query

| foo=bar&lorem=ipsum

param: ref

http ://user:password@www.google.com:80 /command?foo=bar&lorem=ipsum#fragment

```
java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)
| .....
param: protocol
| http .....
param: host
| ..... www.google.com .....
param: port
| ..... 80 .....
param: authority
| .....
param: userinfo
| ..... user:password .....
param: path
| ..... /command .....
param: query
| ..... foo=bar&lorem=ipsum .....
param: ref
| ..... fragment
```


http ://user:password@www.google.com:80 /command?foo=bar&lorem=ipsum#fragment

```
java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)
| .....
param: protocol
| http .....
param: host
| .....www.google.com.....
param: port
| .....80.....
param: authority
| .....user:password@www.google.com:80.....
param: userinfo
| .....user:password.....
param: path
| ...../command.....
param: query
| .....foo=bar&lorem=ipsum.....
param: ref
| .....fragment.....
```

http ://user:password@www.google.com:80 /command?foo=bar&lorem=ipsum#fragment

```
java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)
| http...user:password@www.google.com:80/command•foo=bar&lorem=ipsum•fragment
param: protocol
| http .....
param: host
| .....www.google.com .....
param: port
| .....80 .....
param: authority
| .....user:password@www.google.com:80 .....
param: userinfo
| .....user:password .....
param: path
| ...../command .....
param: query
| .....foo=bar&lorem=ipsum .....
param: ref
| .....fragment
```

```

java.net.URL set(protocol, host, port, authority, userinfo, path, query, ref)
| http user:password@www.google.com:80 /command foo=bar&lorem=ipsum fragment
param: protocol
| http
param: host
| www.google.com
param: port
| 80
param: authority
| user:password@www.google.com:80
param: userinfo
| user:password
param: path
| /command
param: query
| foo=bar&lorem=ipsum
param: ref
| fragment

```

URL ::= PROTOCOL '://' AUTHORITY
 AUTHORITY ::= USERINFO '@' HOST

```

java.net.URL.set(protocol, host, port, authority, userinfo, path, query, ref)
| http...user:password@www.google.com:80/command•foo=bar&lorem=ipsum•fragment
param: protocol
| http.....
param: host
| .....www.google.com.....
param: port
| .....80.....
param: authority
| .....user:password@www.google.com:80.....
param: userinfo
| .....user:password.....
param: path
| ...../command.....
param: query
| .....foo=bar&lorem=ipsum.....
param: ref
| .....fragment

```



```

URL ::= PROTOCOL '://' AUTHORITY PATH '?' QUERY '#' REF
AUTHORITY ::= USERINFO '@' HOST ':' PORT
PROTOCOL ::= 'http'
USERINFO ::= 'user:password'
HOST ::= 'www.google.com'
PORT ::= '80'
PATH ::= '/command'
QUERY ::= 'foo=bar&lorem=ipsum'
REF ::= 'fragment'

```

URLs

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`
`http://www.guardian.co.uk/sports/worldcup#results`
`ftp://bob:12345@ftp.example.com/oss/debian7.iso`



```
URL ::= PROTOCOL '://' AUTHORITY PATH ['?' QUERY] ['#' REF]
AUTHORITY ::= [USERINFO '@'] HOST [':' PORT]
PROTOCOL ::= 'http' | 'ftp'
USERINFO ::= /[a-z]+:[a-z]+/
HOST ::= /[a-z.]+/
PORT ::= '80'
PATH ::= /\[/[a-z0-9.\//]*\//
QUERY ::= 'foo=bar&lorem=ipsum'
REF ::= /[a-z]+/
```

URLs

```
URL ::= PROTOCOL '://' AUTHORITY PATH ['?' QUERY] ['#' REF]
AUTHORITY ::= [USERINFO '@'] HOST [':' PORT]
PROTOCOL ::= 'http' | 'ftp'
USERINFO ::= /[a-z]+:[a-z]+/
HOST ::= /[a-z.]+/
PORT ::= '80'
PATH ::= /\[/[a-z0-9.\//]*\//
QUERY ::= 'foo=bar&lorem=ipsum'
REF ::= /[a-z]+/
```



```
http://6F35:PkT5v@2.5/,,
http://.g:8
http://C.Ta.2./p.,//1.#14cq5
http://.37...g:776/,,
http://.:07//,.8B,#eUN027
http://87.:2117//?&=&&38#207
http://S1t26c:7223i@.1...:16207
ftp://wb428:lr@00.8y.#5W7V9U2
ftp://012304.xt9Ut@k.285?250===K
```

http://mE:26Ciu@.8...:1528/8,,2,,?===&r&
ftp://rW:L@0H....:8111/7,,g/,
http://D...C
http://2.6j0:032277
http://x1f0...:332334?&==2==&
http://3u8Wabn:tN@m:3592#36
ftp://2.8...:9161208/..?=&=9#5F
ftp://.n:7945457//?9
http://Jy:98/9,3?===q
http://G42:7Nz596e@6.4b//F/,?&I=0
ftp://.697..?===SU=
http://3d00:ud@.1dF9/2q//5
ftp://.d5...8:646#D
ftp://62ql1:40P63@4.:321727?=
http://.//,,/
ftp://8zN3xl:3499l8@t036./,3?=&=40
http://B7j85D3:NvPd7M@.8.p.:5/,,,#e7JS
http://t4...:124///6,G.?=&&=#3F2Qx
http://YP6:zKG@.:946775?=#Zb7
http://./,31,,F.#693
ftp://7V:c4748C2@.//...?&&&&2R
http://...:40123?=r=&7I
ftp://.74:4773362/./A#Et
ftp://67:3g5YNi@.5M.2...:06716?&=#3W758V6
ftp://i:cqj97@..2..3:362287?&=&&7f5#4
http://1:l@N..6..i:667//,,6,
http://70o0:518@3:4791089#962
ftp://zA35Qsu:56@..5...:997/,,
ftp://8.../5?&n#7i1C7G3
ftp://2:fm0@J.:6208/,Z/H#3GZ747b
http://2:7p54n14@8r09.1
ftp://XK3438:w169KkU@..5R.8?=6g

INI Files

```
[Application]
Version = 0.5
WorkingDir = /tmp/mydir/
[User]
User = Bob
Password = 12345
```



```
INI ::= LINE+
LINE ::= SECTION_LINE '\r'
        | OPTION_LINE  ['\r']
SECTION_LINE ::= '[' KEY ']'
OPTION_LINE  ::= KEY ' = ' VALUE
KEY ::= /[a-zA-Z]*/
VALUE ::= /[a-zA-Z0-9\]/
```


JSON Files

<JSON files>



```
JSON ::= WHITESPACE VALUE WHITESPACE
WHITESPACE ::= /[ \n\t]+/
VALUE ::= JSONOBJECT | ARRAY | STRING |
          TRUE | FALSE | NULL | NUMBER
TRUE ::= 'true'
FALSE ::= 'false'
NULL ::= 'null'
NUMBER ::= ['-'] /[0-9]+/
STRING ::= '"' INTERNALSTRING '"'
INTERNALSTRING ::= /[a-zA-Z0-9 ]+/
ARRAY ::=
'['
  [WHITESPACE]
  [VALUE [WHITESPACE]
    [',' [WHITESPACE] VALUE [WHITESPACE]]+]
  ']'
JSONOBJECT ::=
'{'
  [WHITESPACE]
  [STRING [WHITESPACE] ':' [WHITESPACE]
    VALUE [WHITESPACE]
    [',' [WHITESPACE]
      STRING [WHITESPACE] ':' [WHITESPACE]
      VALUE [WHITESPACE]]
  +]
'{'
```

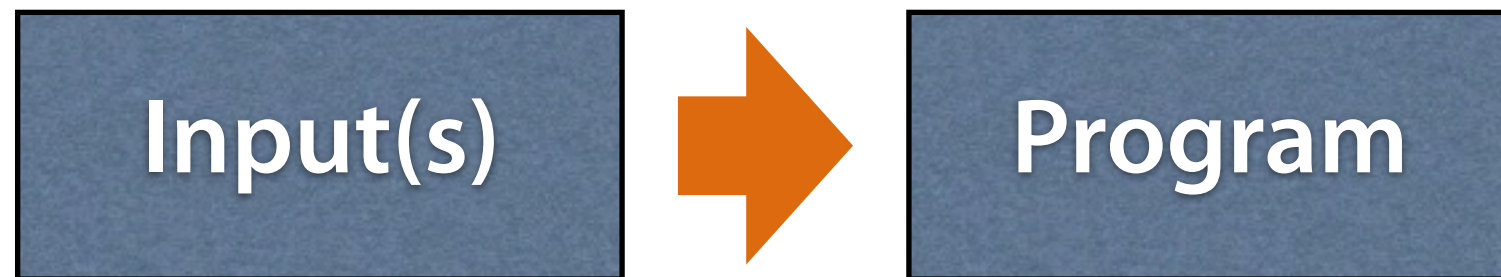
Evaluation

Subject	Accuracy	Completeness
<code>java.lang.Url</code>	82,3 %	100 %
Apache Commons CSV	100 %	100 %
<code>java.util.Properties</code>	100 %	100 %
INI4J	64,6 %	100 %
Minimal JSON	100 %	100 %

Challenges

- Table based parsers
- Binary formats
- Context sensitive features
- Multi-layer input formats
- Learning grammars without samples

Program Behavior

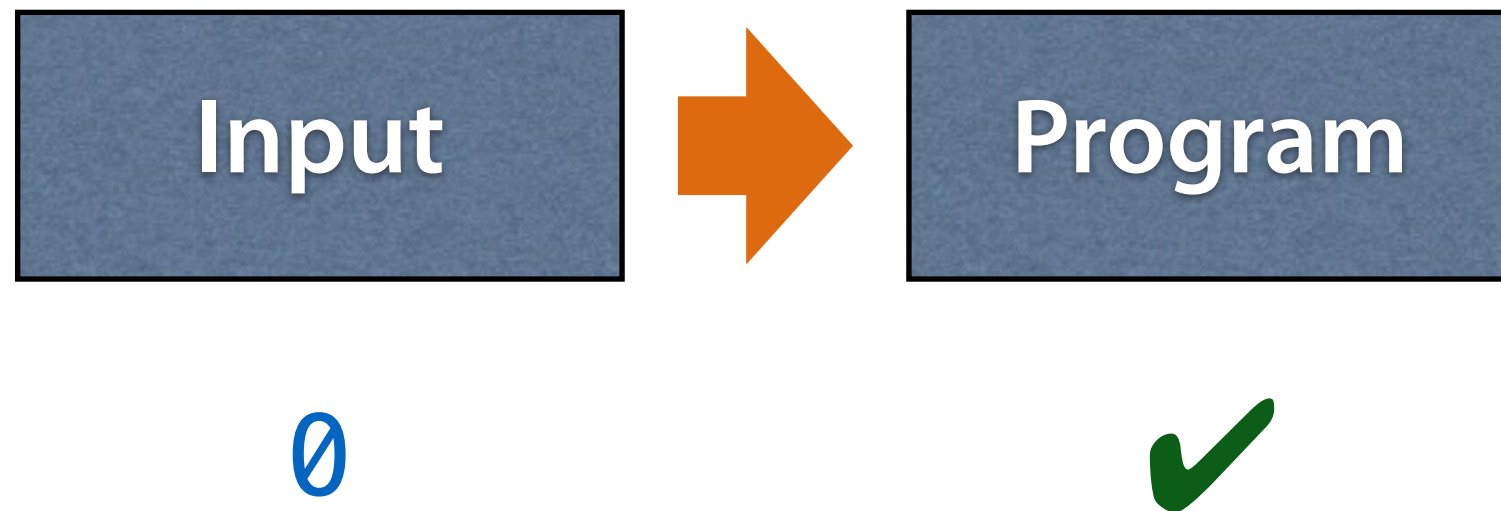


xyzzzy

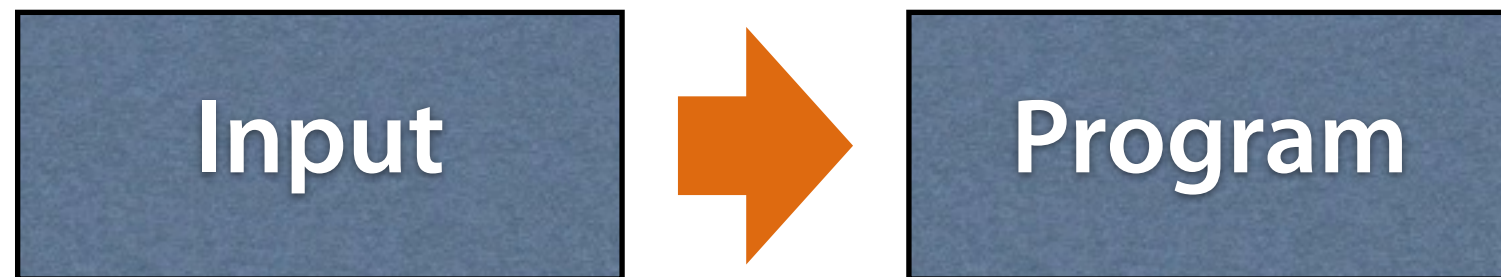


- checks for digit
- checks for "true"/"false"
- checks for ""
- checks for '['
- checks for '{'

Program Behavior



Program Behavior

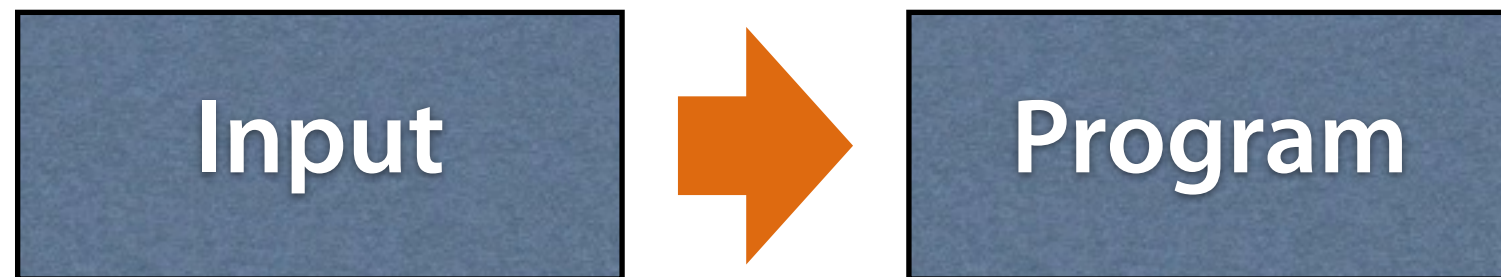


0



- checks for digit
- checks for "true"/"false"
- checks for ""
- checks for '['
- checks for '{'

Program Behavior

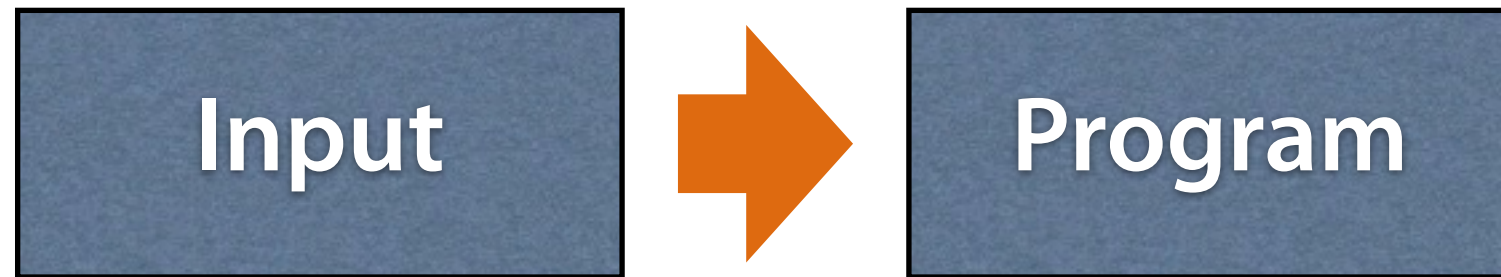


0



- checks for digit
- checks for "true"/"false"
- checks for ""
- checks for '['
- checks for '{'

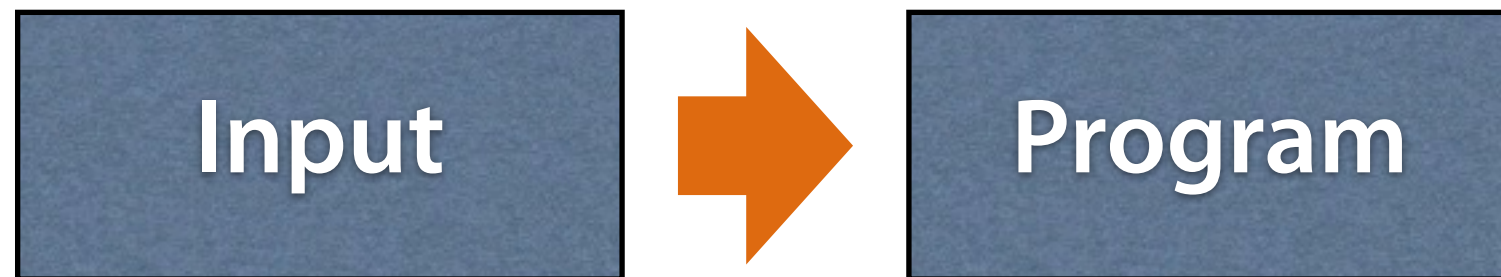
Program Behavior



||

X

Program Behavior

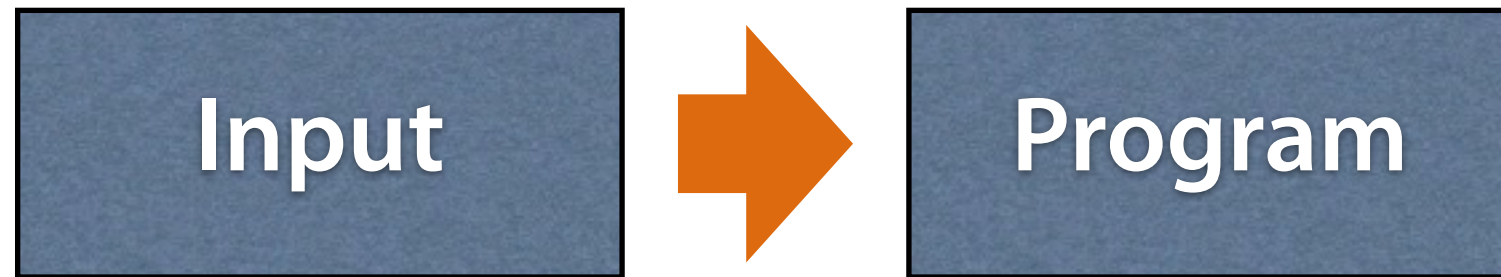


||



- checks for `""`
- checks for `'\'`
- checks for character

Program Behavior



||||



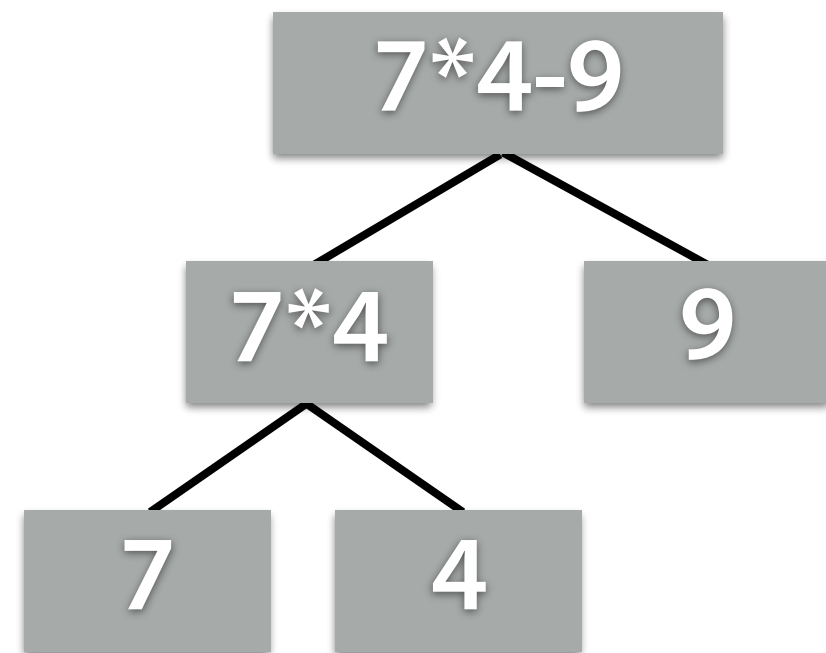
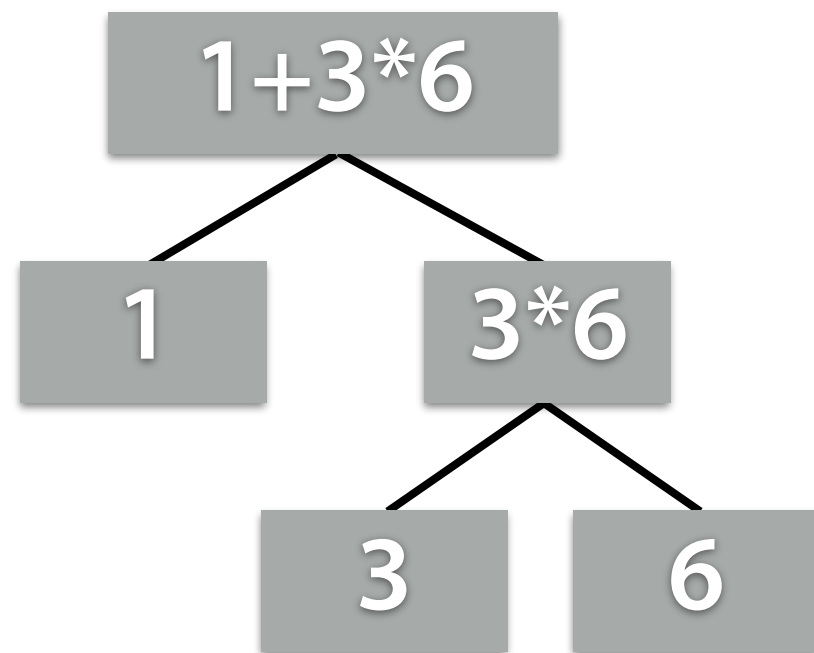
JSON Files (no samples)

<JSON files>



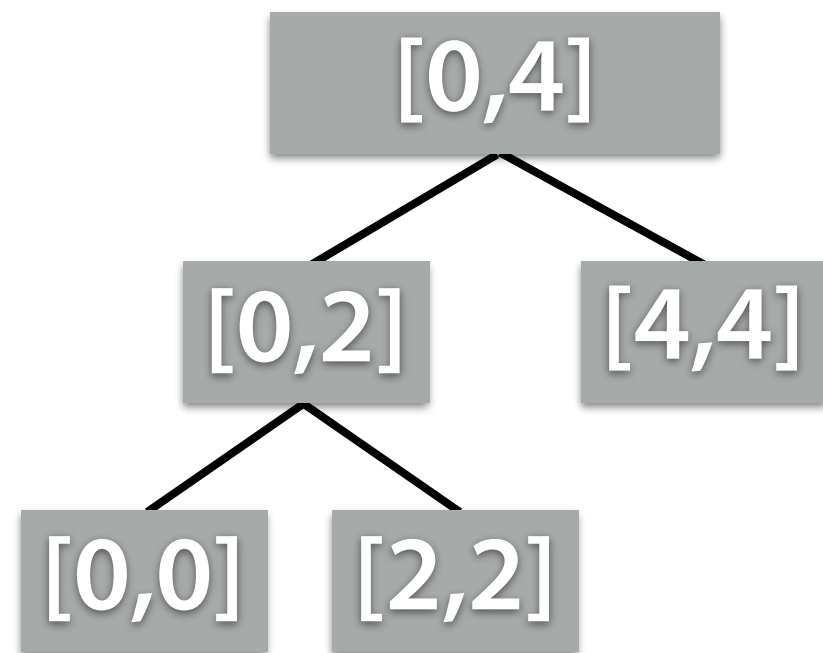
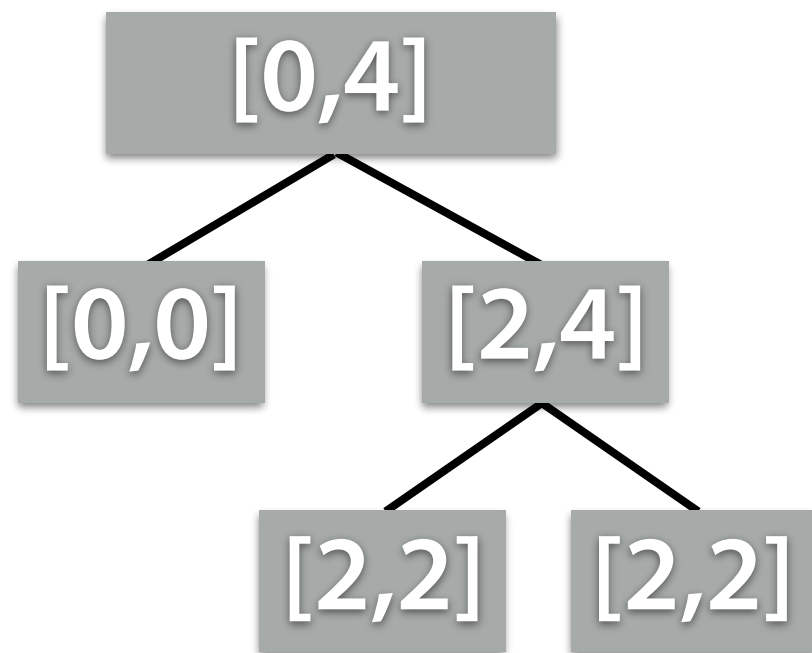
```
JSON ::= VALUE
VALUE ::= OBJECT | ARRAY | STRING |
        TRUE | FALSE | NULL | NUMBER
OBJECT ::= '{' [NAME ':' VALUE [ '¶' ] ] '}'
ARRAY  ::= '[' [VALUE [',' VALUE [ '¶' ] ] +] ']'
STRING ::= "\"\\"" | "'" ENDCAPTURE [ "'" ]
ENDCAPTURE ::= ('e' | 'E' | 's' | '0' | "'" )+
TRUE ::= 'true'
FALSE ::= 'false'
NULL  ::= 'null'
NAME  ::= "\"\\""
NUMBER ::= /[0-9]+/ [ FRACTION | EXPONENT ]
FRACTION ::= ".0"
EXPONENT ::= ('e' | 'E' | '+' | '-' | '0' )+
```

Implementation



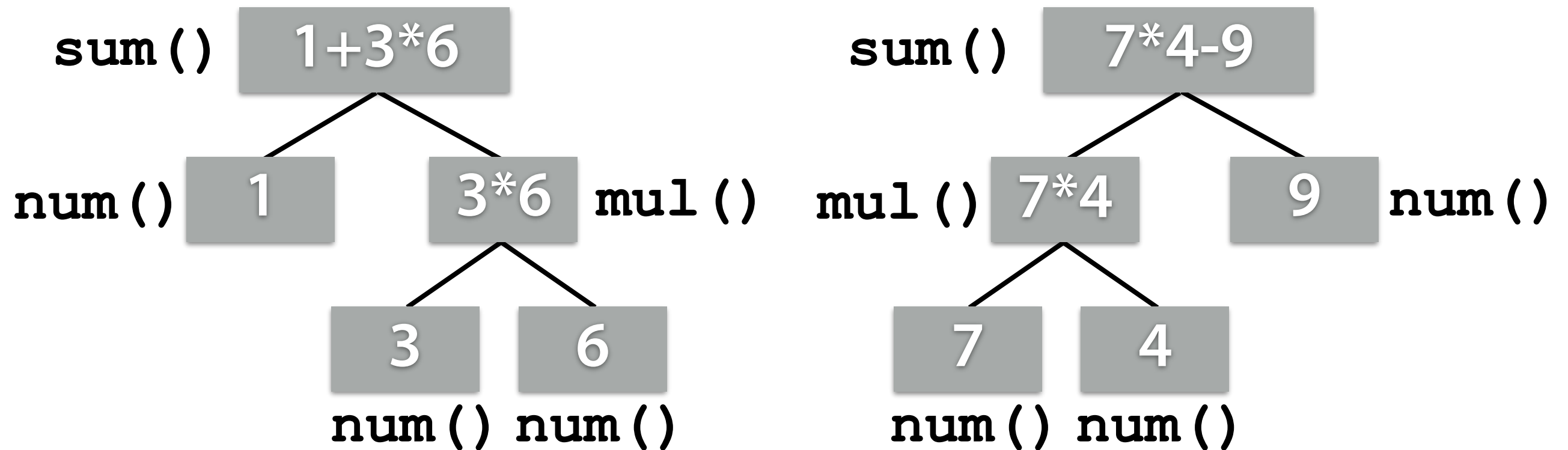
Implementation

Interval Trees



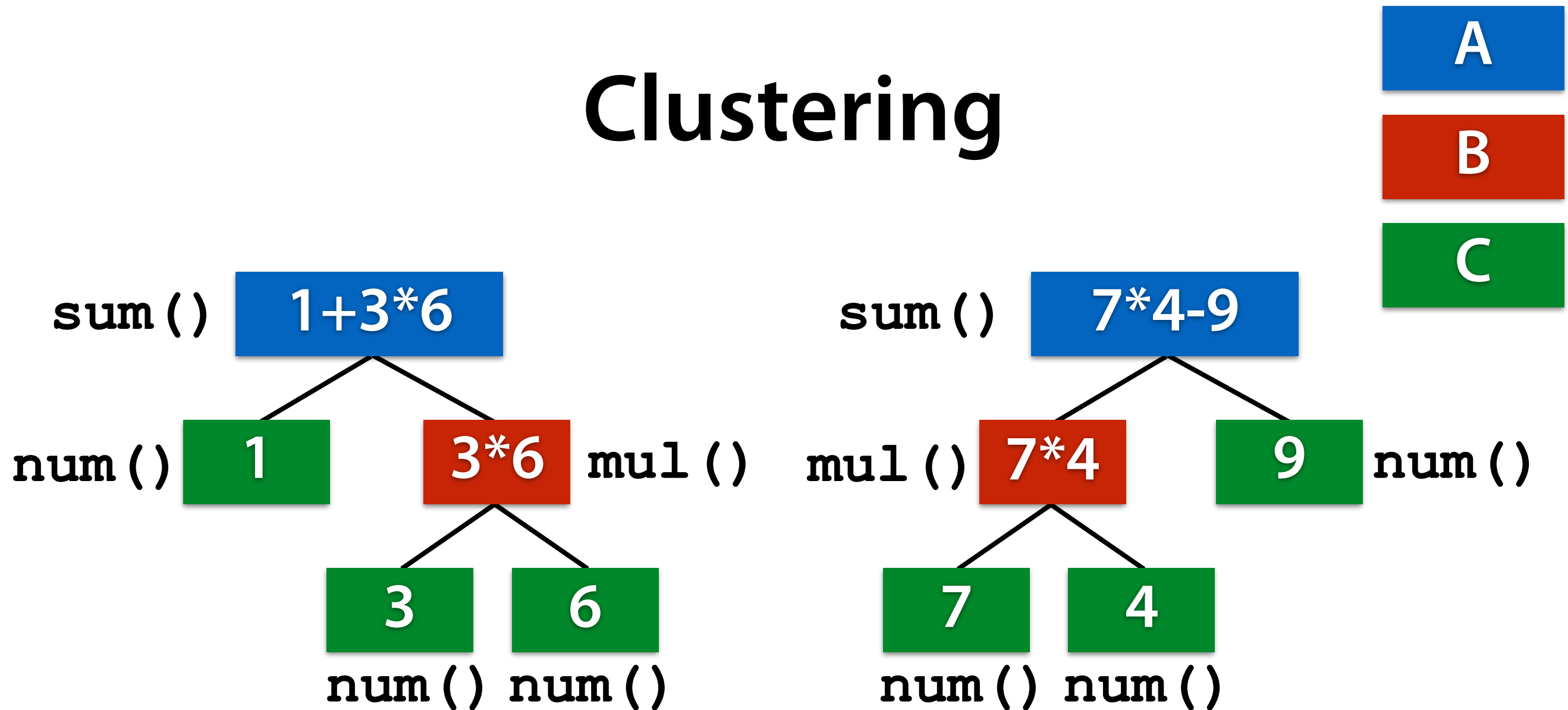
Implementation

Data Flow



Implementation

Clustering



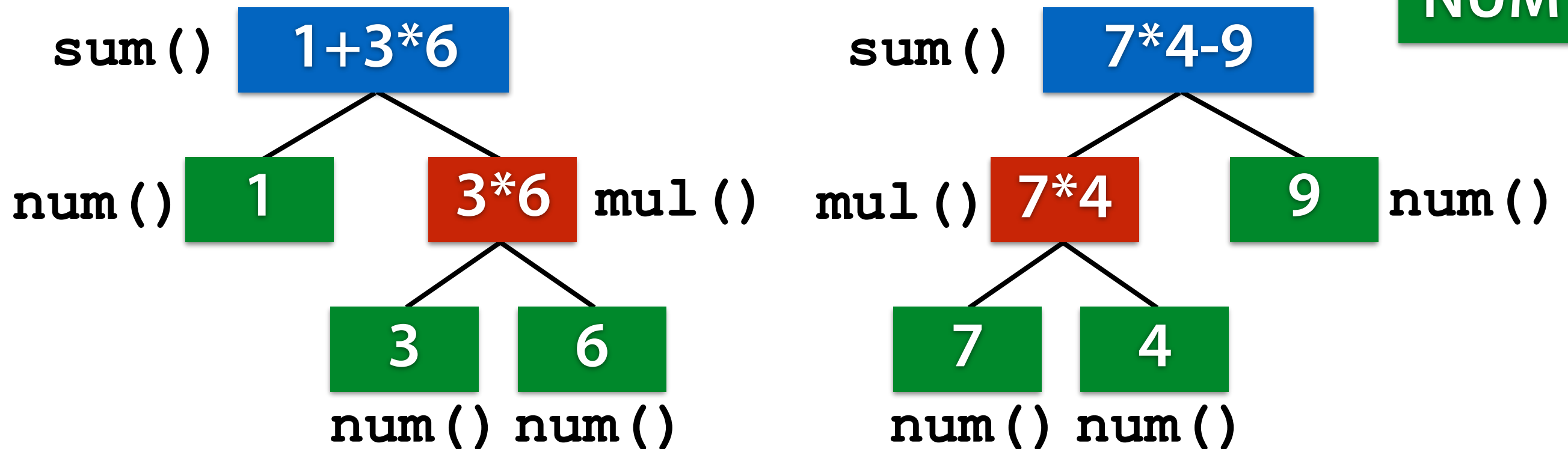
Implementation

Naming

SUM

MUL

NUM



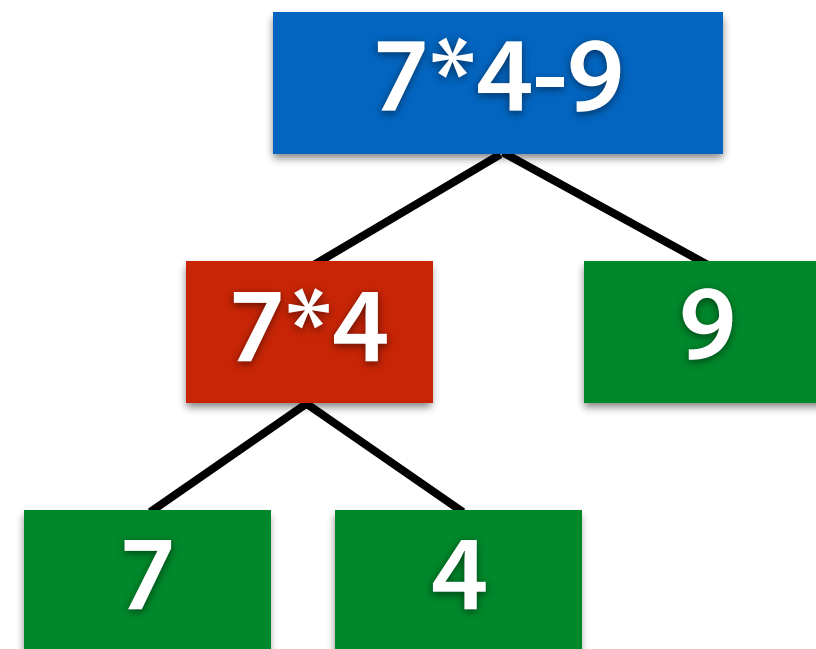
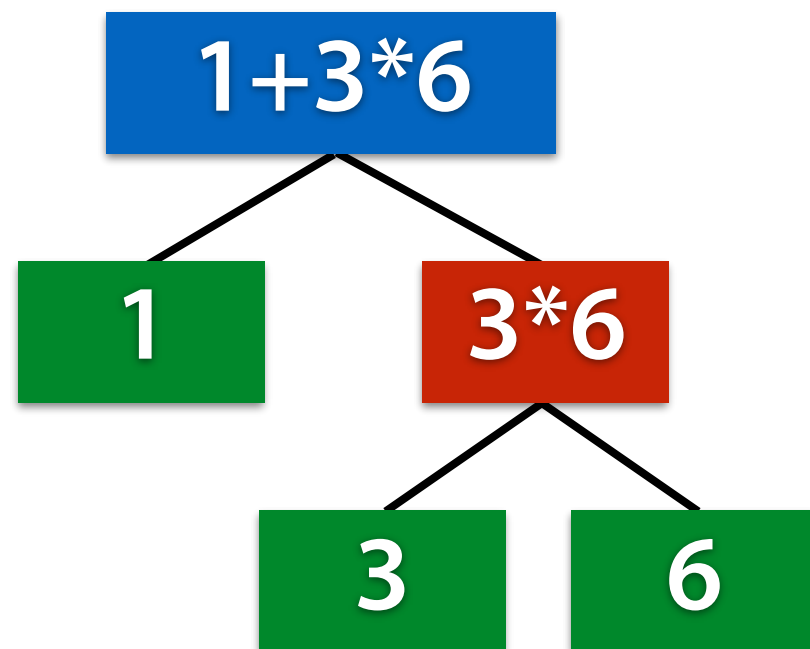
Implementation

Productions

SUM

MUL

NUM



Implementation

Productions

SUM	:=	1+3*6		7*4-9								
MUL	:=	3*6		7*4								
NUM	:=	1		3		4		6		7		9

Implementation

Productions

SUM	:=	NUM	+	MUL		MUL	-	NUM				
MUL	:=	NUM	*	NUM								
NUM	:=	1		3		4		6		7		9

Implementation

Productions

Start: **SUM**

SUM	:=	NUM	+	MUL		MUL	-	NUM				
MUL	:=	NUM	*	NUM								
NUM	:=	1		3		4		6		7		9

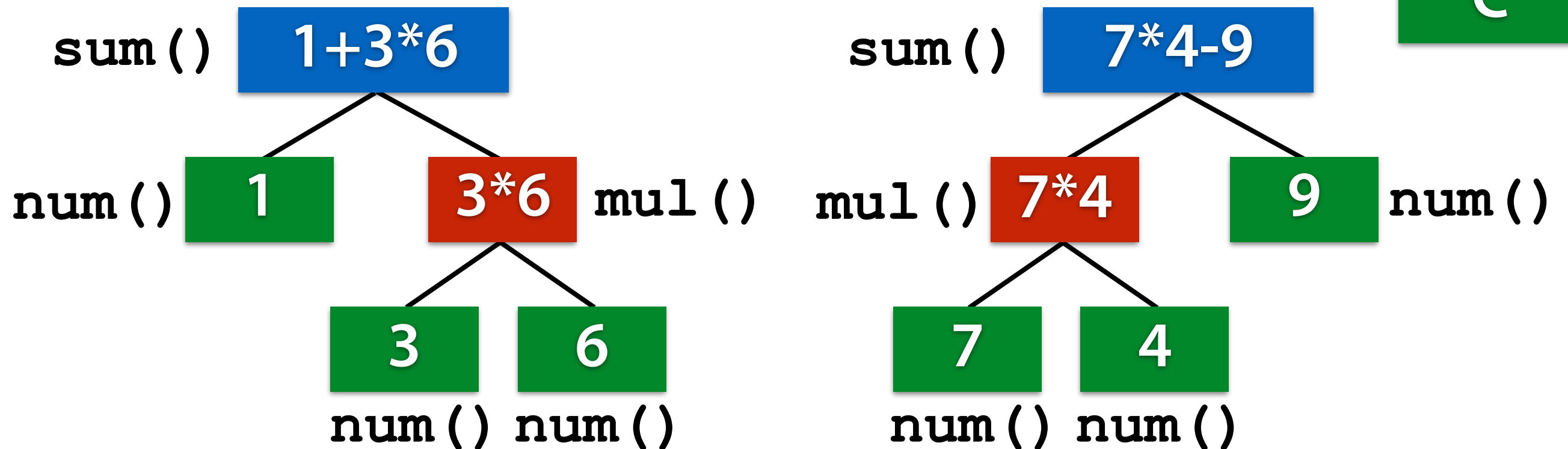
Implementation In Python

Clustering: ParseElement

A

B

C



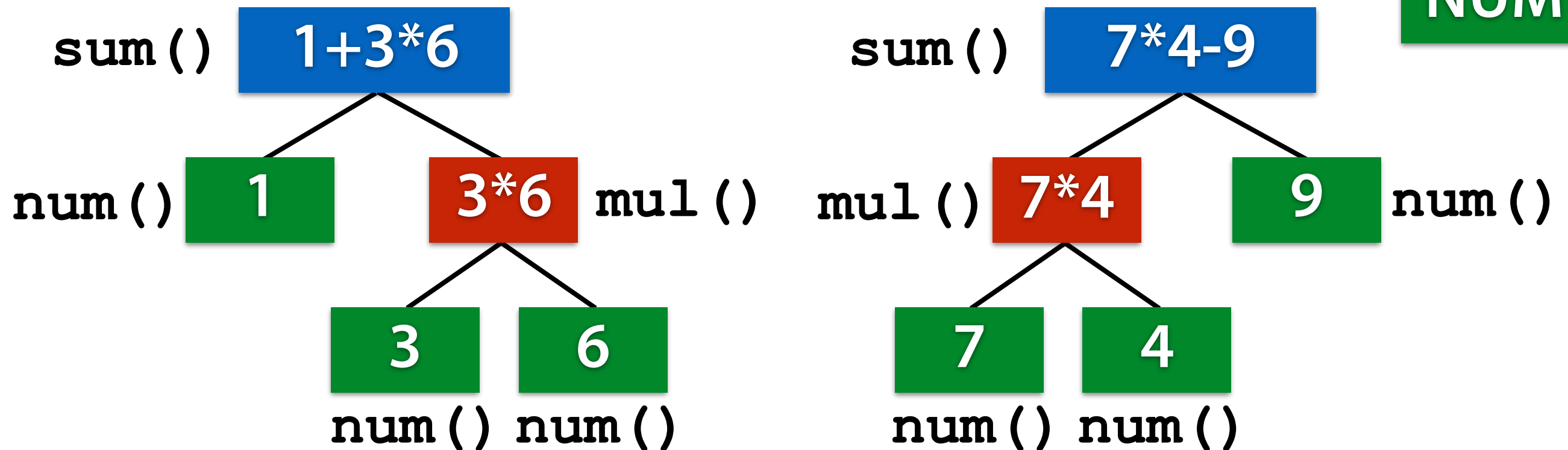
Implementation In Python

Naming: Variables in BNF ()

SUM

MUL

NUM



Implementation In Python

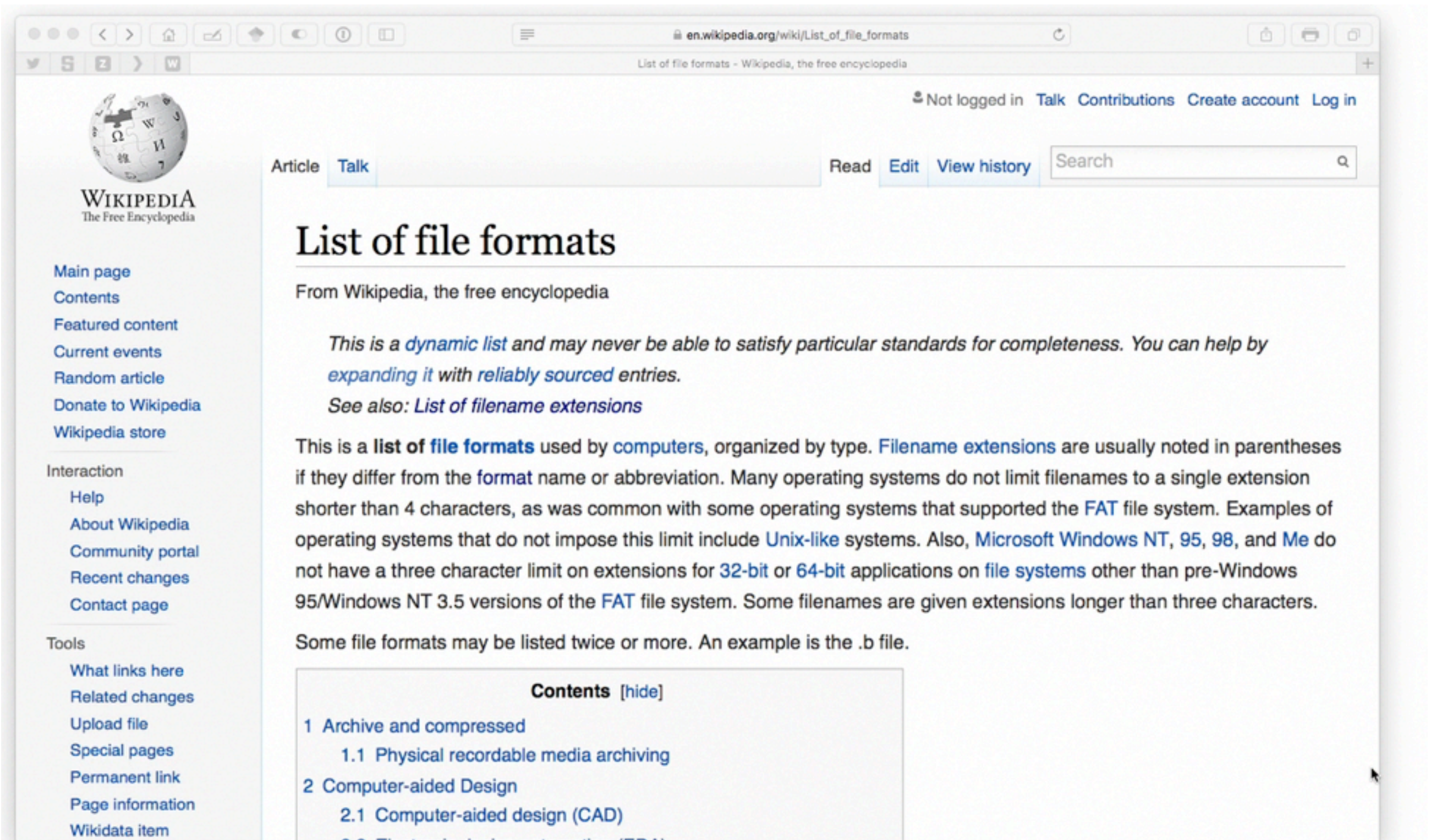
Productions: Subclasses of ParseElement

Start: **SUM**

SUM	:=	NUM	+	MUL		MUL	-	NUM				
MUL	:=	NUM	*	NUM								
NUM	:=	1		3		4		6		7		9

Coding Demo

File Formats



The image is a screenshot of a web browser displaying the Wikipedia article titled "List of file formats". The browser's address bar shows the URL "en.wikipedia.org/wiki/List_of_file_formats". The page layout includes a sidebar on the left with navigation links such as "Main page", "Contents", "Featured content", "Current events", "Random article", "Donate to Wikipedia", and "Wikipedia store". The main content area features the article title "List of file formats" and a sub-header "From Wikipedia, the free encyclopedia". A notice states: "This is a *dynamic list* and may never be able to satisfy particular standards for completeness. You can help by *expanding it with reliably sourced entries*." Below this, it says "See also: *List of filename extensions*". The article text explains that this is a list of file formats used by computers, organized by type, with filename extensions noted in parentheses. It mentions that many operating systems do not limit filenames to a single extension shorter than 4 characters, as was common with some systems supporting the FAT file system. Examples of systems that do not impose this limit include Unix-like systems, and Microsoft Windows NT, 95, 98, and Me. It also notes that some systems do not have a three-character limit on extensions for 32-bit or 64-bit applications on file systems other than pre-Windows 95/Windows NT 3.5 versions of the FAT file system. Some filenames are given extensions longer than three characters. The article concludes by stating that some file formats may be listed twice or more, with the .b file given as an example. At the bottom, there is a "Contents" section with a "[hide]" link, listing the following sections: 1 Archive and compressed, 1.1 Physical recordable media archiving, 2 Computer-aided Design, 2.1 Computer-aided design (CAD), and 2.2 Electronic design automation (EDA).

en.wikipedia.org/wiki/List_of_file_formats

List of file formats - Wikipedia, the free encyclopedia

Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search

List of file formats

From Wikipedia, the free encyclopedia

*This is a **dynamic list** and may never be able to satisfy particular standards for completeness. You can help by **expanding it with reliably sourced entries**.*

*See also: **List of filename extensions***

This is a **list of file formats** used by **computers**, organized by type. **Filename extensions** are usually noted in parentheses if they differ from the **format** name or abbreviation. Many operating systems do not limit filenames to a single extension shorter than 4 characters, as was common with some operating systems that supported the **FAT** file system. Examples of operating systems that do not impose this limit include **Unix-like** systems. Also, **Microsoft Windows NT**, **95**, **98**, and **Me** do not have a three character limit on extensions for **32-bit** or **64-bit** applications on **file systems** other than pre-Windows 95/Windows NT 3.5 versions of the **FAT** file system. Some filenames are given extensions longer than three characters.

Some file formats may be listed twice or more. An example is the .b file.

Contents [hide]

- 1 Archive and compressed
 - 1.1 Physical recordable media archiving
- 2 Computer-aided Design
 - 2.1 Computer-aided design (CAD)
 - 2.2 Electronic design automation (EDA)

Learned Grammars

- give insights into the *structure of inputs*
 - reverse engineering
 - writing tests
 - writing parsers
- can directly be used for *test generators*
 - automated fuzzing tools

Mining Input Grammars

