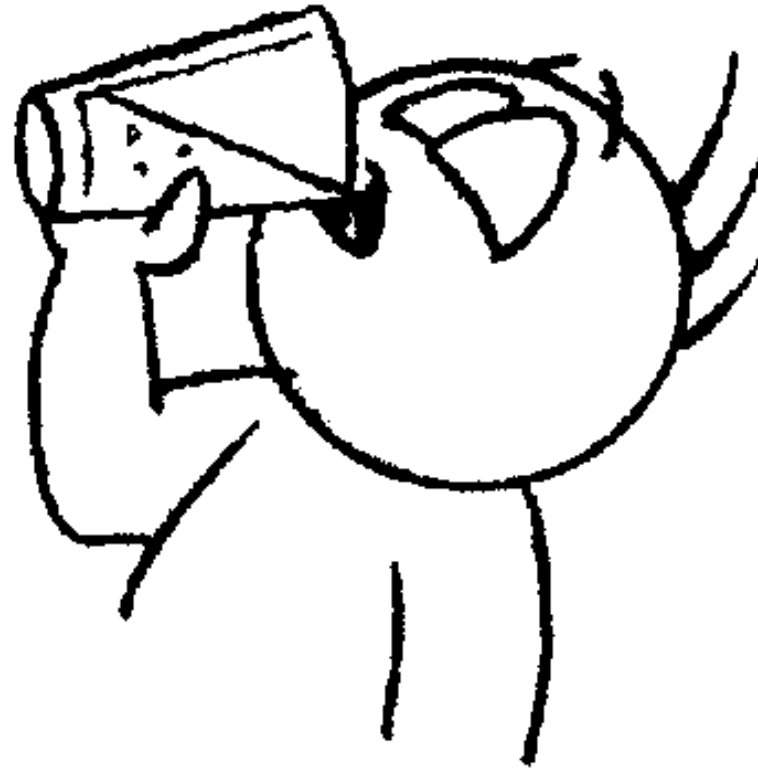


Bottom-Up Parsing

What is Bottom-Up Parsing?

- Idea: Apply productions **in reverse** to convert the user's program to the start symbol.
- As with top-down, could be done with a DFS or BFS, though this is rarely done in practice.
- We'll be exploring four **directional, predictive** bottom-up parsing techniques:
 - **Directional**: Scan the input from left-to-right.
 - **Predictive**: Guess which production should be inverted.

Bottoms Up!



One View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

<code>int</code>	<code>+</code>	<code>(</code>	<code>int</code>	<code>+</code>	<code>int</code>	<code>+</code>	<code>int</code>	<code>)</code>	<code>\$</code>
------------------	----------------	----------------	------------------	----------------	------------------	----------------	------------------	----------------	-----------------

One View of a Bottom-Up Parse

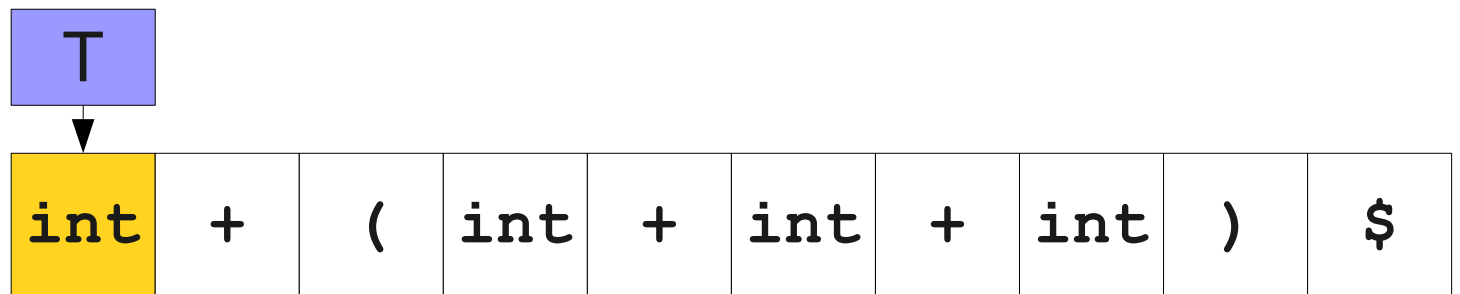
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

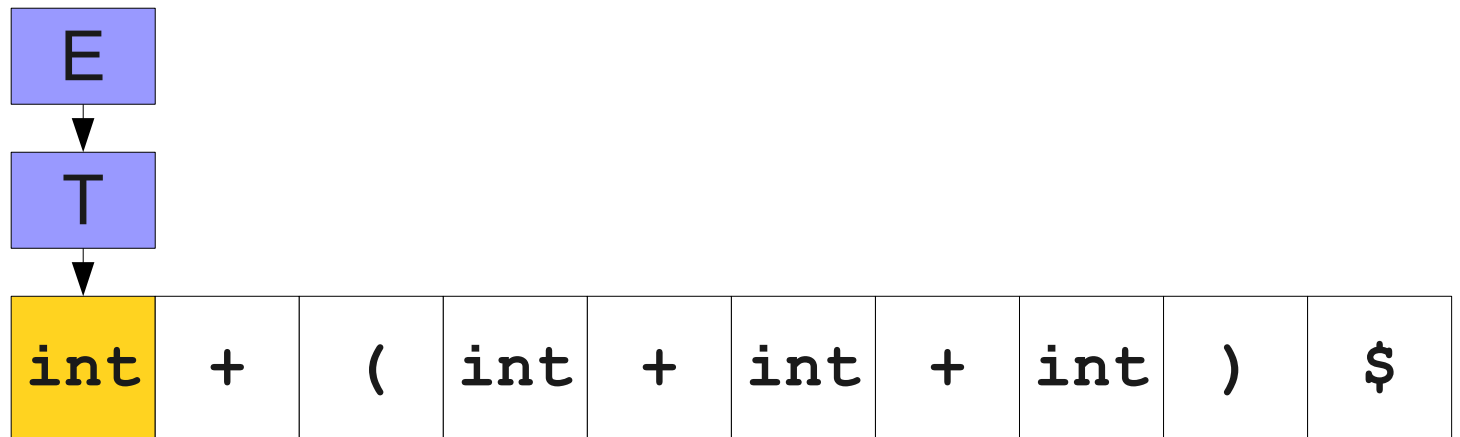
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

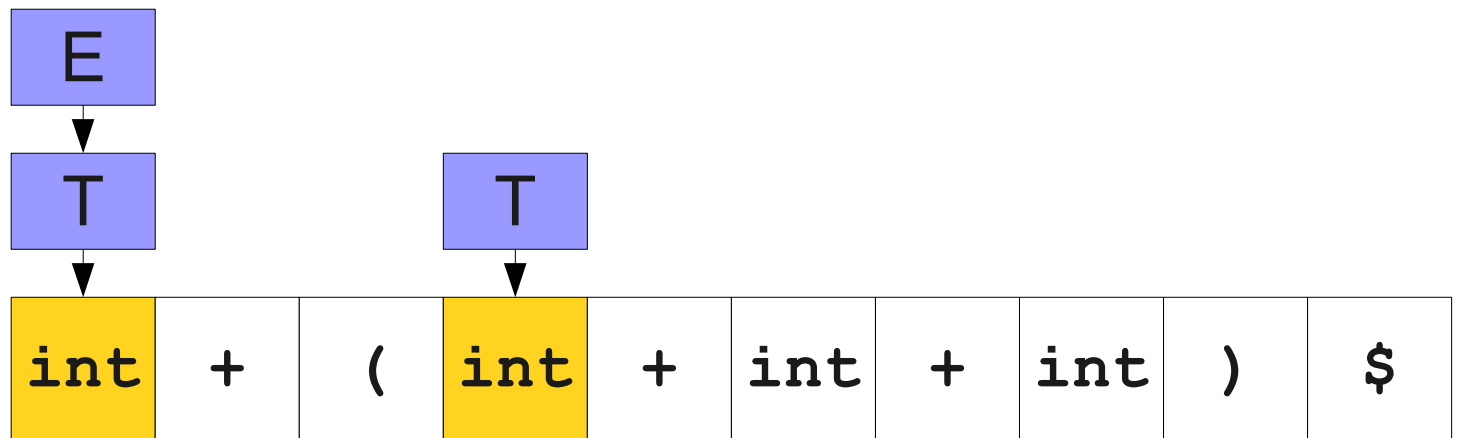
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

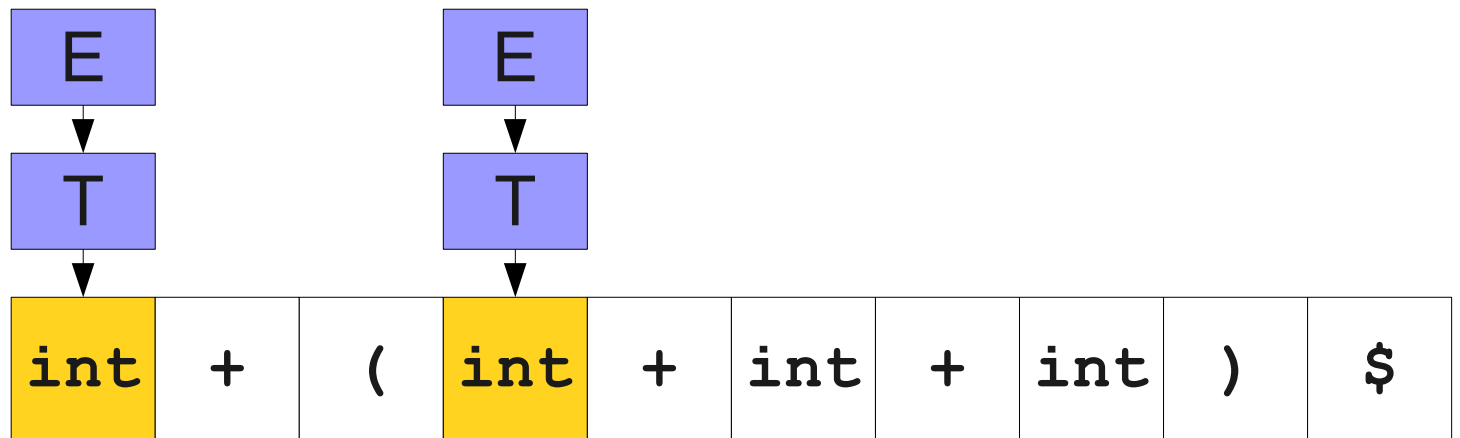
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

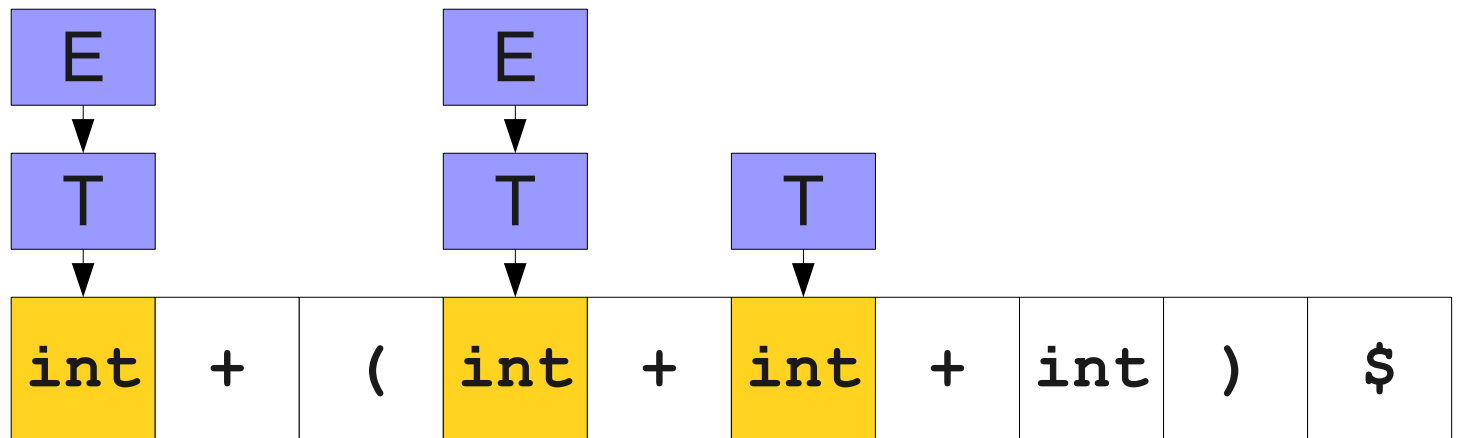
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

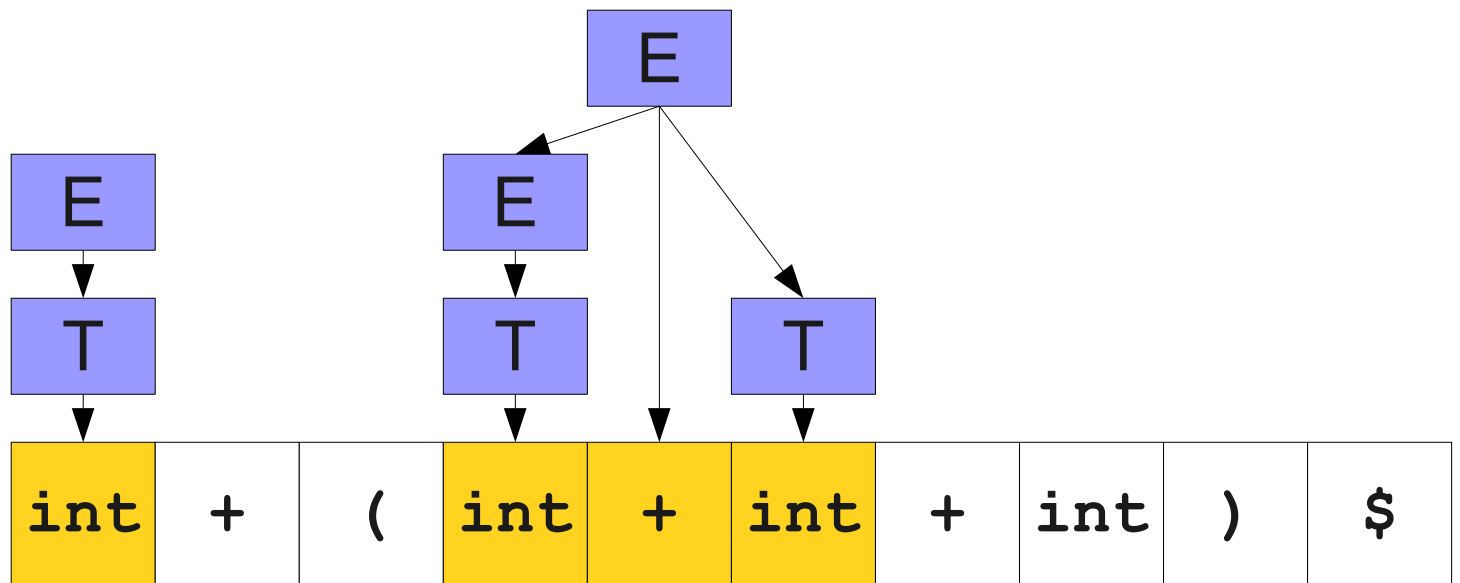
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

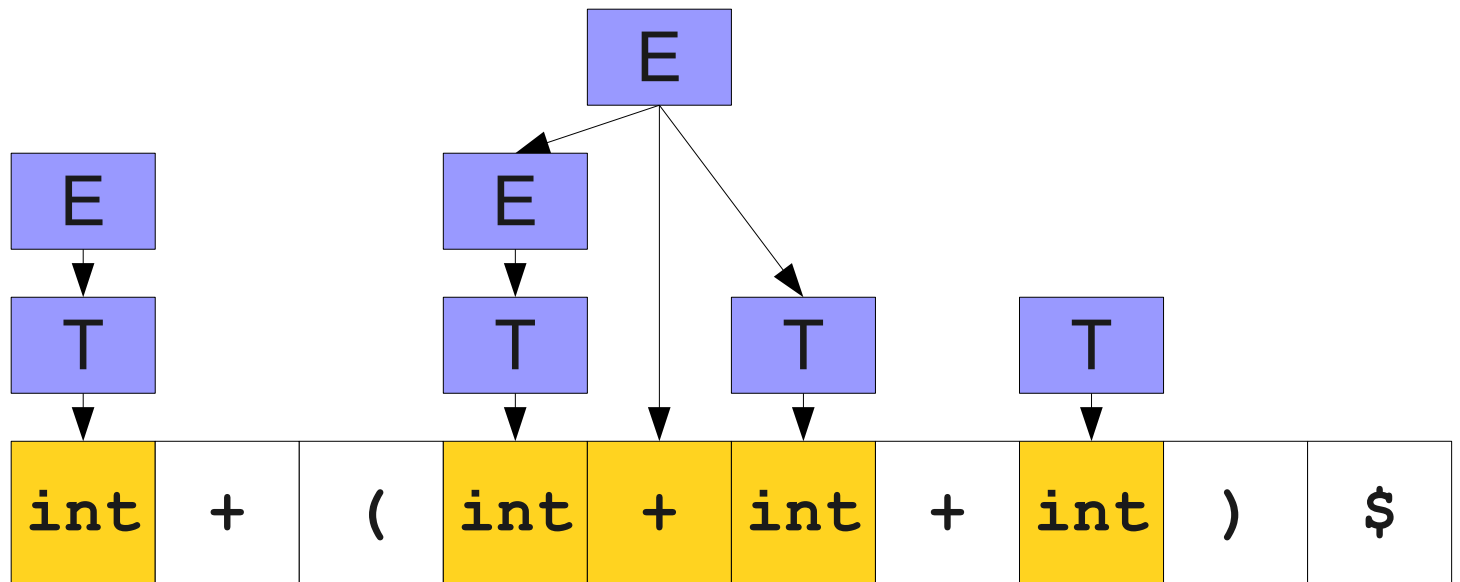
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

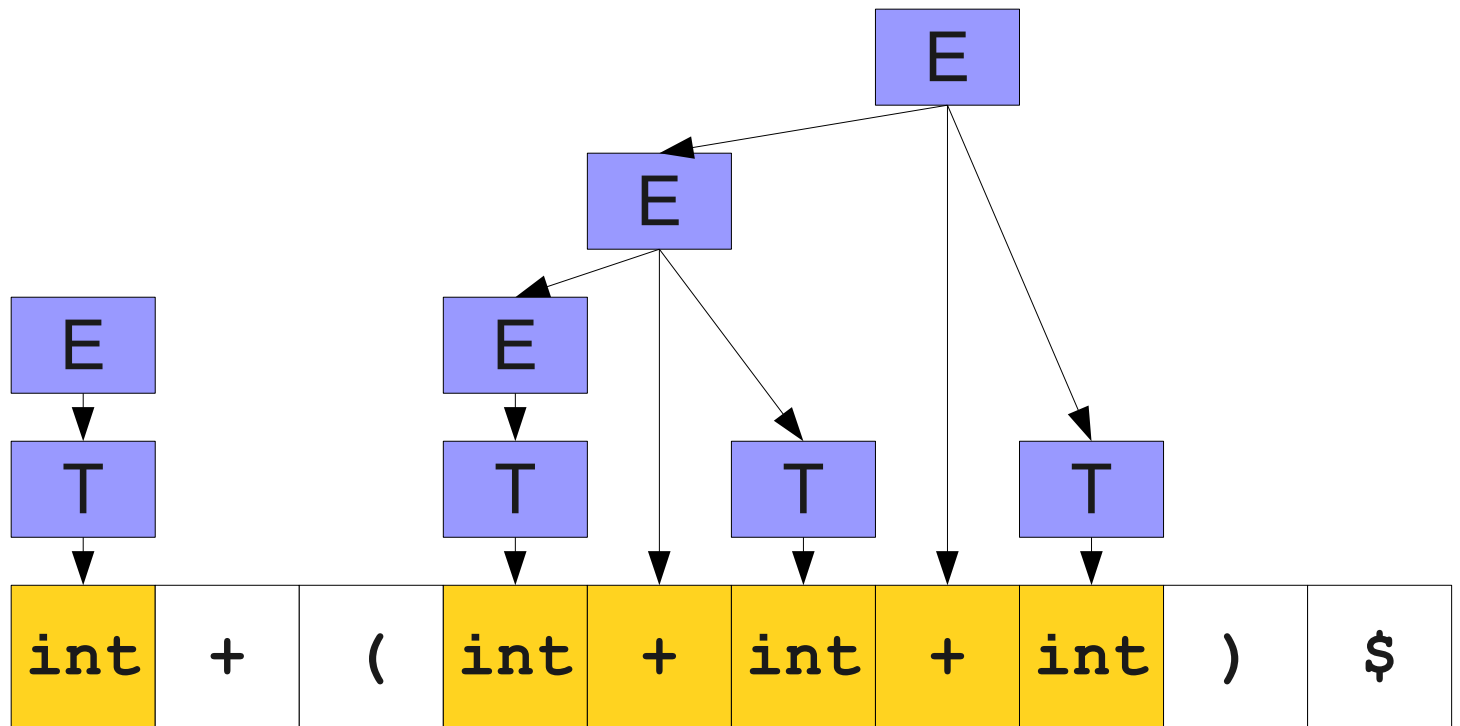
$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

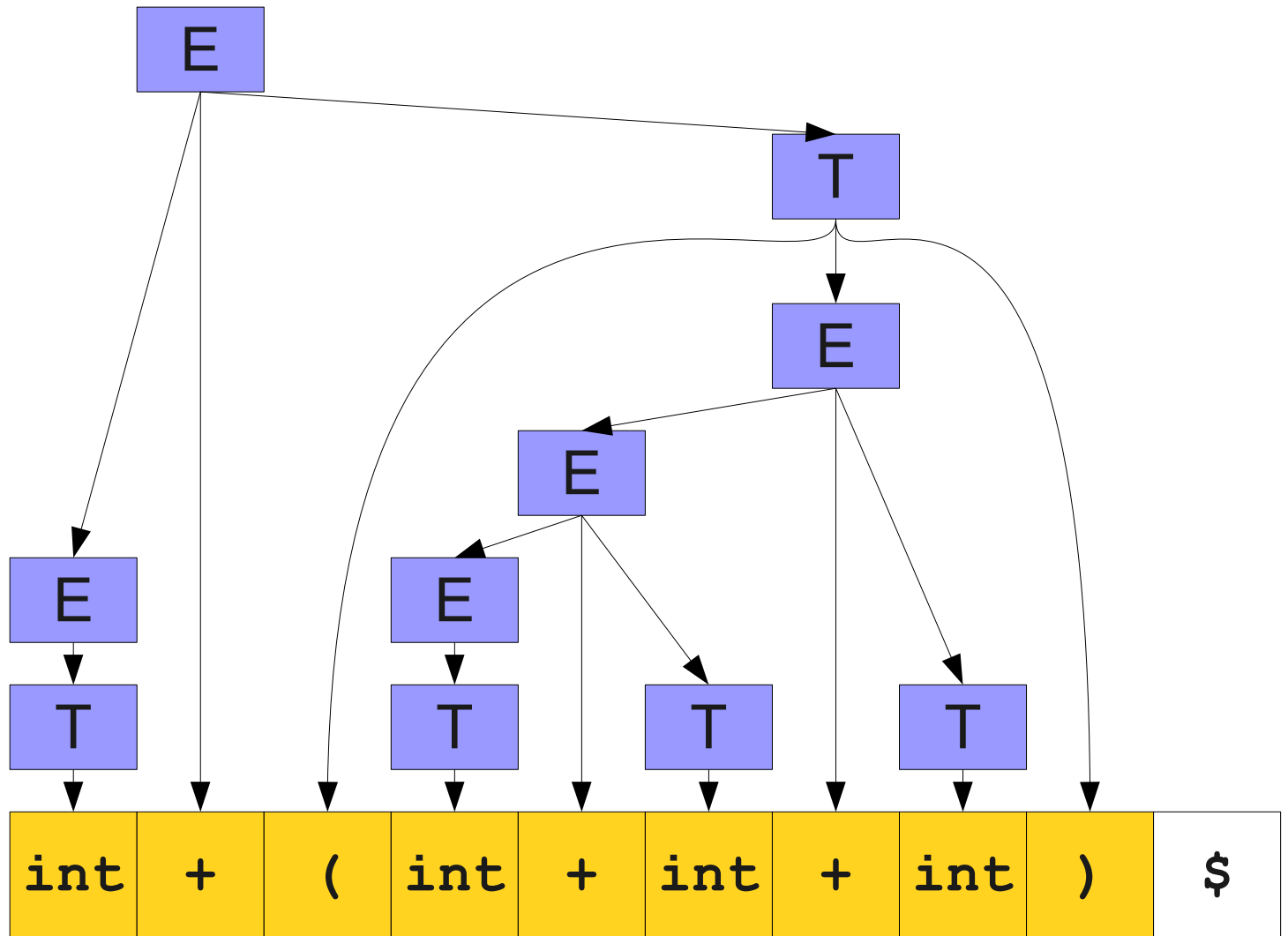
$T \rightarrow \text{int}$

$T \rightarrow (E)$



One View of a Bottom-Up Parse

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$
E	+	(E			+	T)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$
E	+	(E			+	T)	\$
E	+	(E)	\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$
E	+	(E			+	T)	\$
E	+	(E)	\$
E	+	T						\$	

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$
E	+	(E			+	T)	\$
E	+	(E)	\$
E	+	T						\$	
E									\$

A Second View of a Bottom-Up Parse

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$
E	+	(E			+	T)	\$
E	+	(E)	\$
E	+	T						\$	
E									\$
S									

A Second View of a Bottom-Up Parse

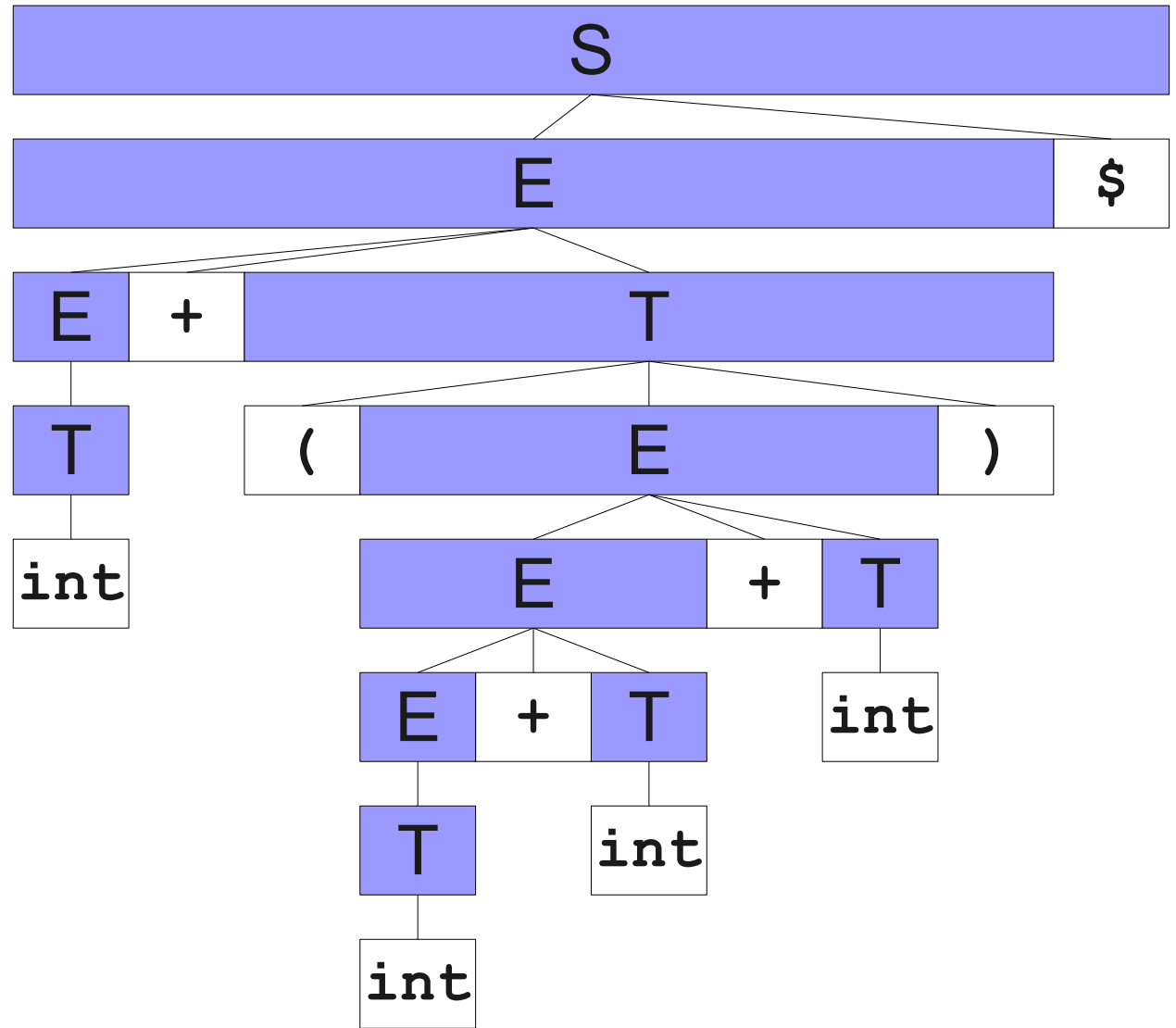
$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

int	+	(int	+	int	+	int)	\$
T	+	(int	+	int	+	int)	\$
E	+	(int	+	int	+	int)	\$
E	+	(T	+	int	+	int)	\$
E	+	(E	+	int	+	int)	\$
E	+	(E	+	T	+	int)	\$
E	+	(E			+	int)	\$
E	+	(E			+	T)	\$
E	+	(E)	\$
E	+	T						\$	
E									\$
S									

A left-to-right, bottom-up parse is a **rightmost derivation traced in reverse.**

A Third View of a Bottom-Up Parse

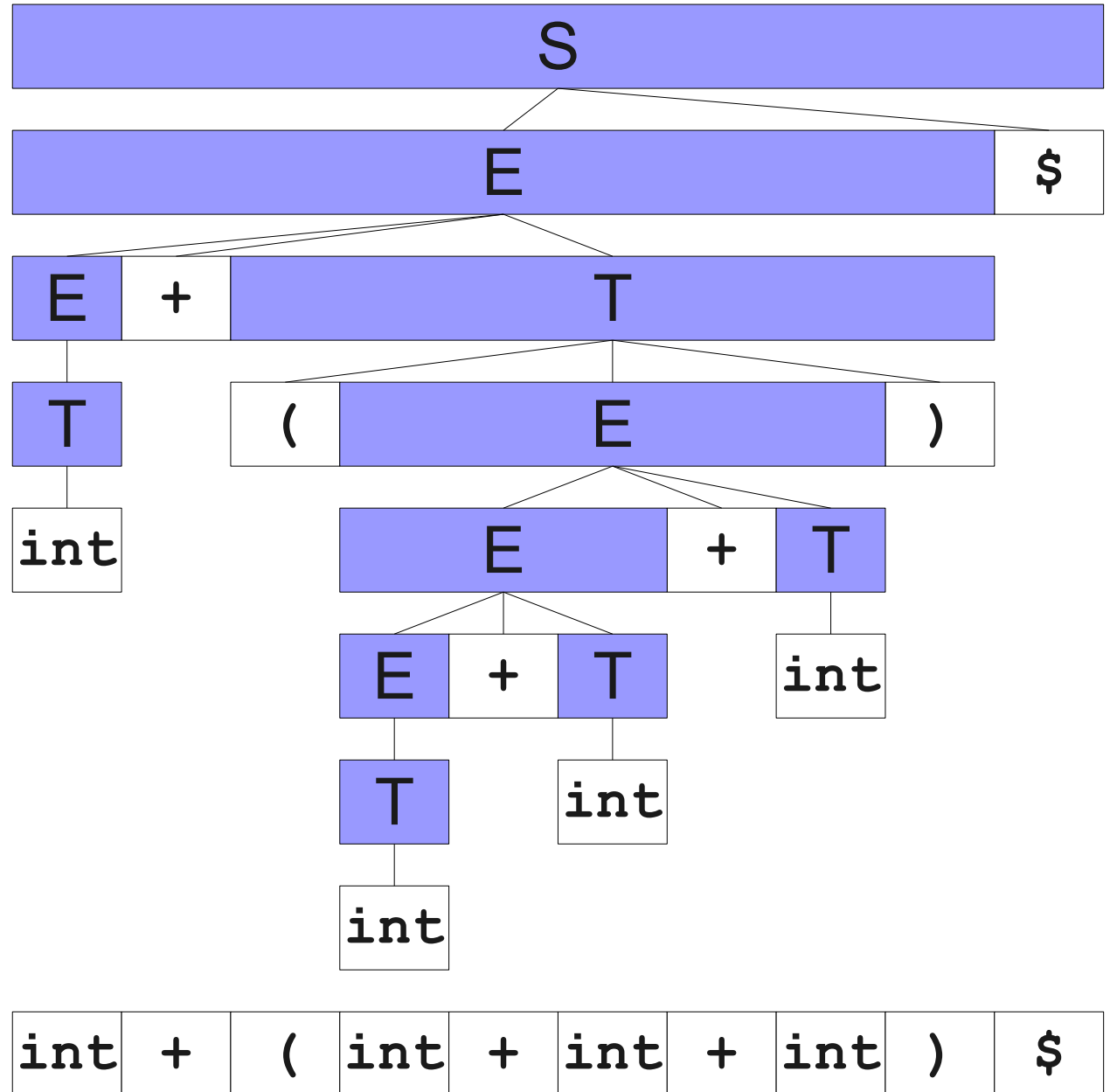
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



`int + (int + int + int) $`

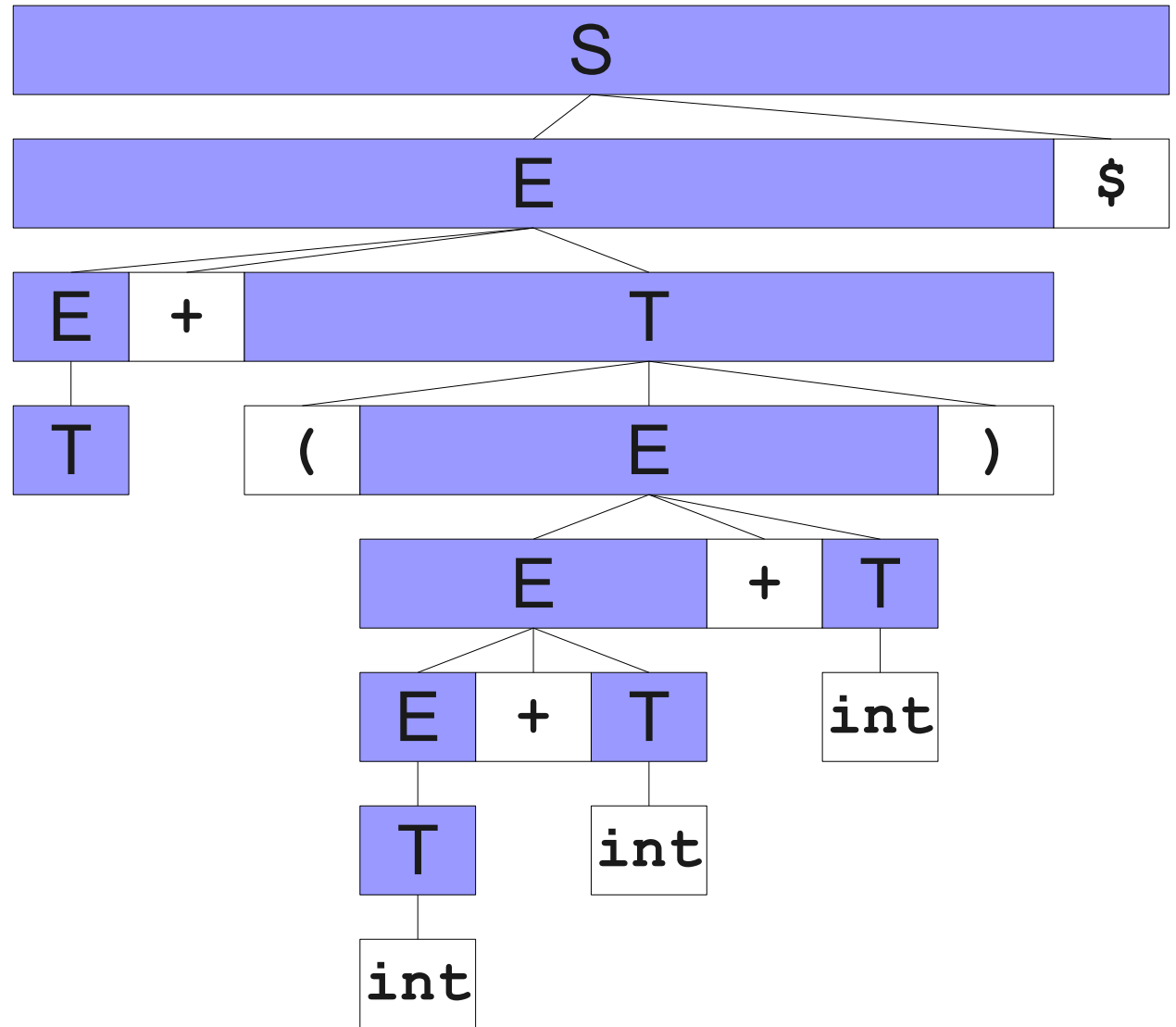
A Third View of a Bottom-Up Parse

`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



A Third View of a Bottom-Up Parse

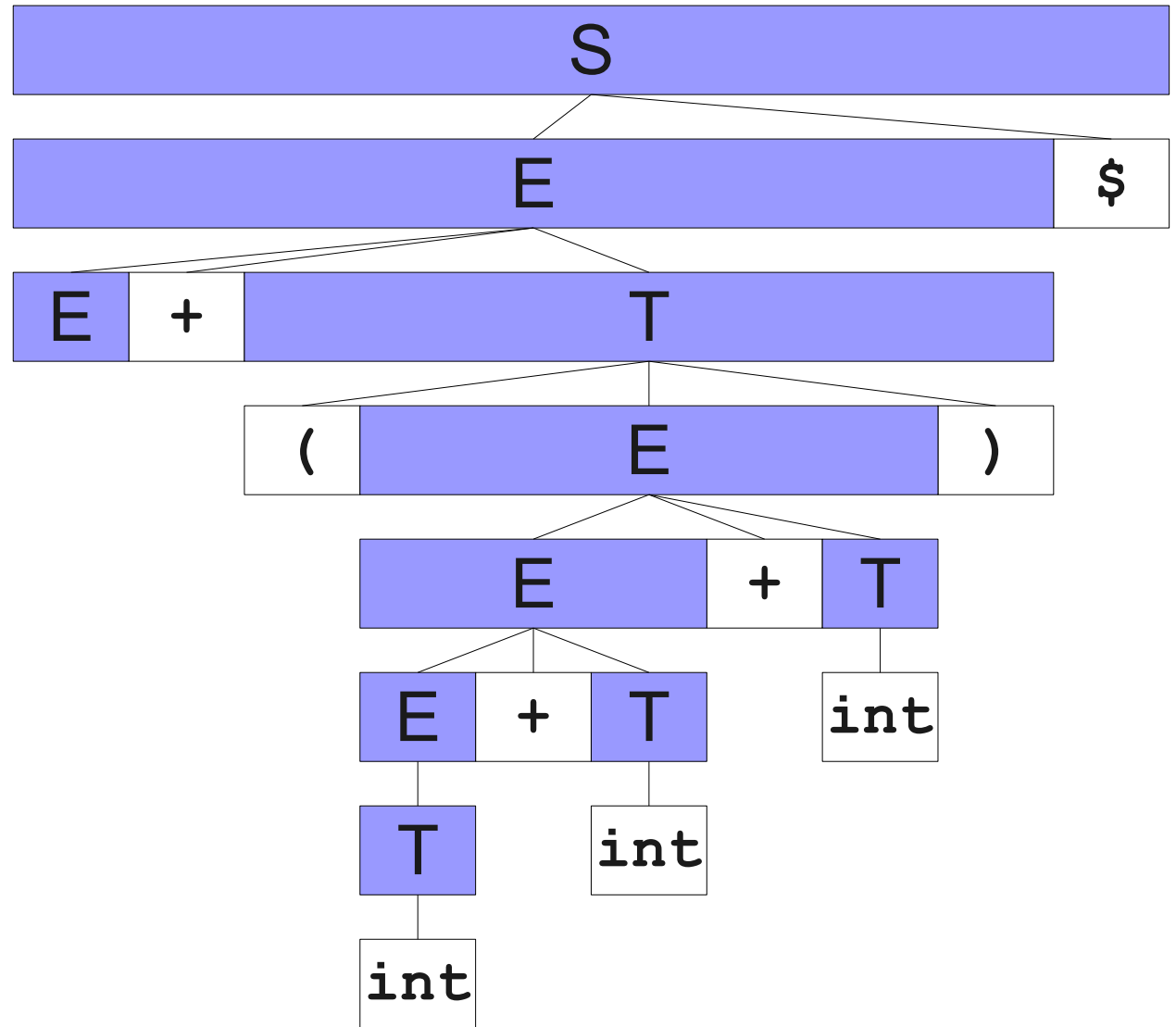
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

A Third View of a Bottom-Up Parse

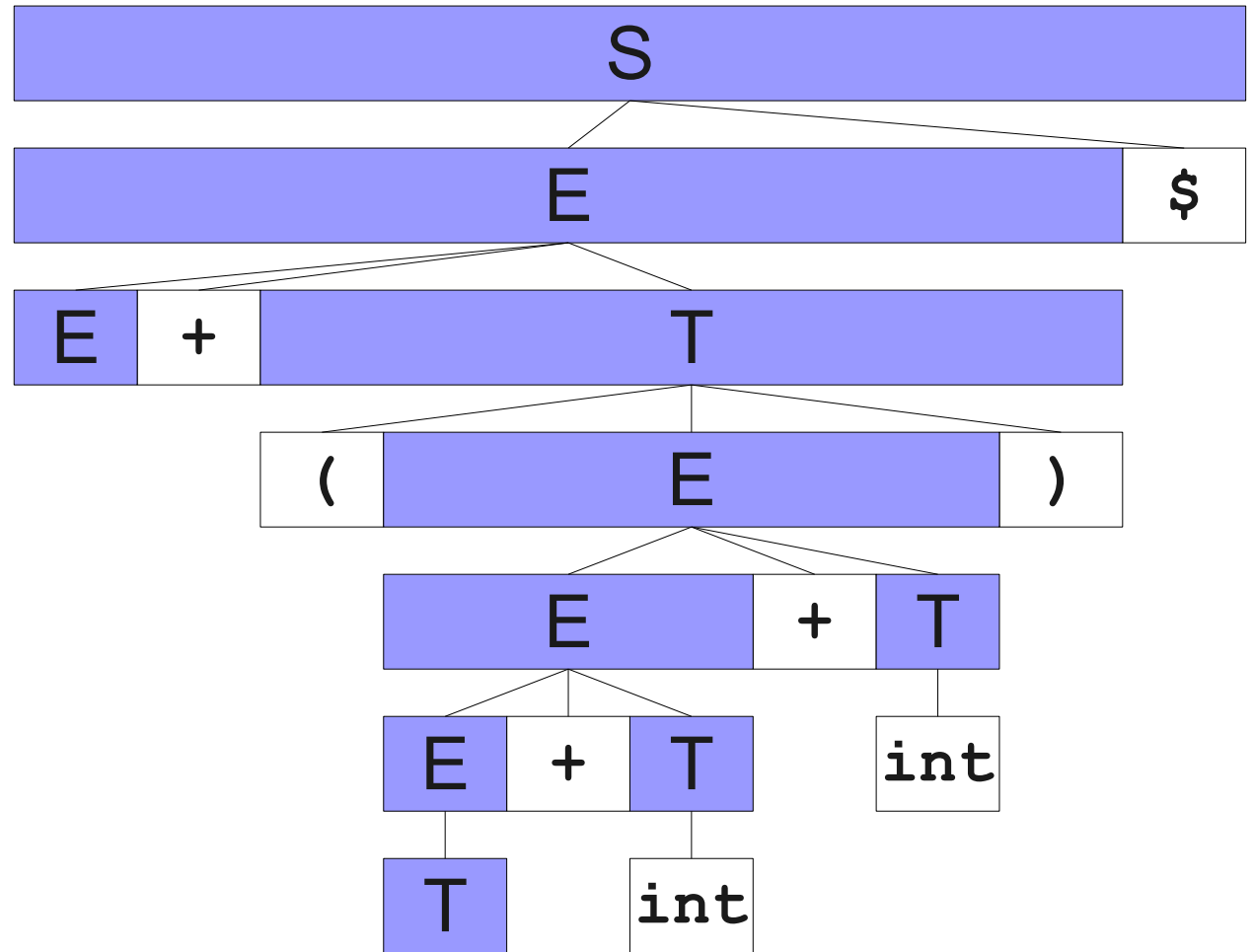
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



`int + (int + int + int) $`

A Third View of a Bottom-Up Parse

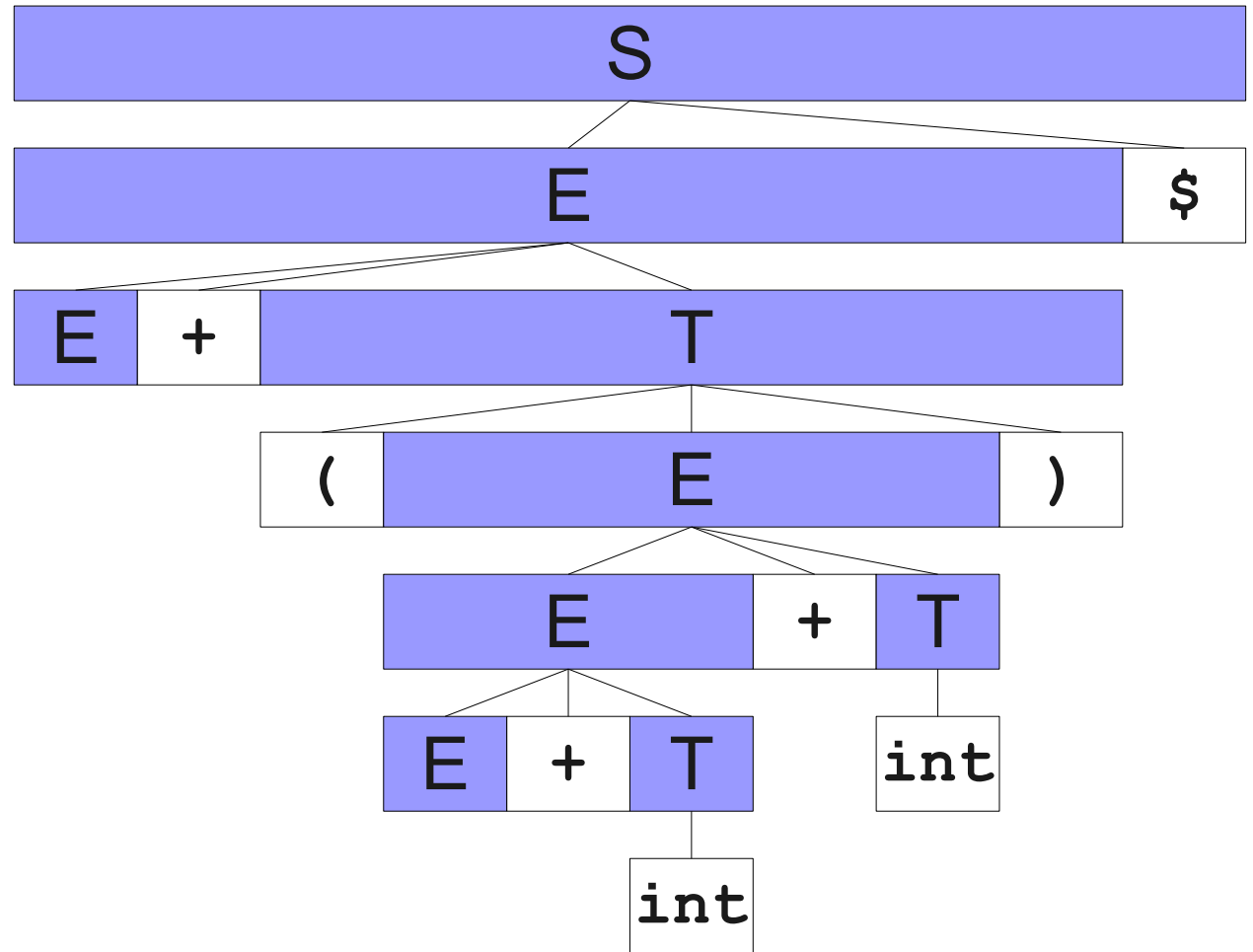
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



`int + (int + int + int) $`

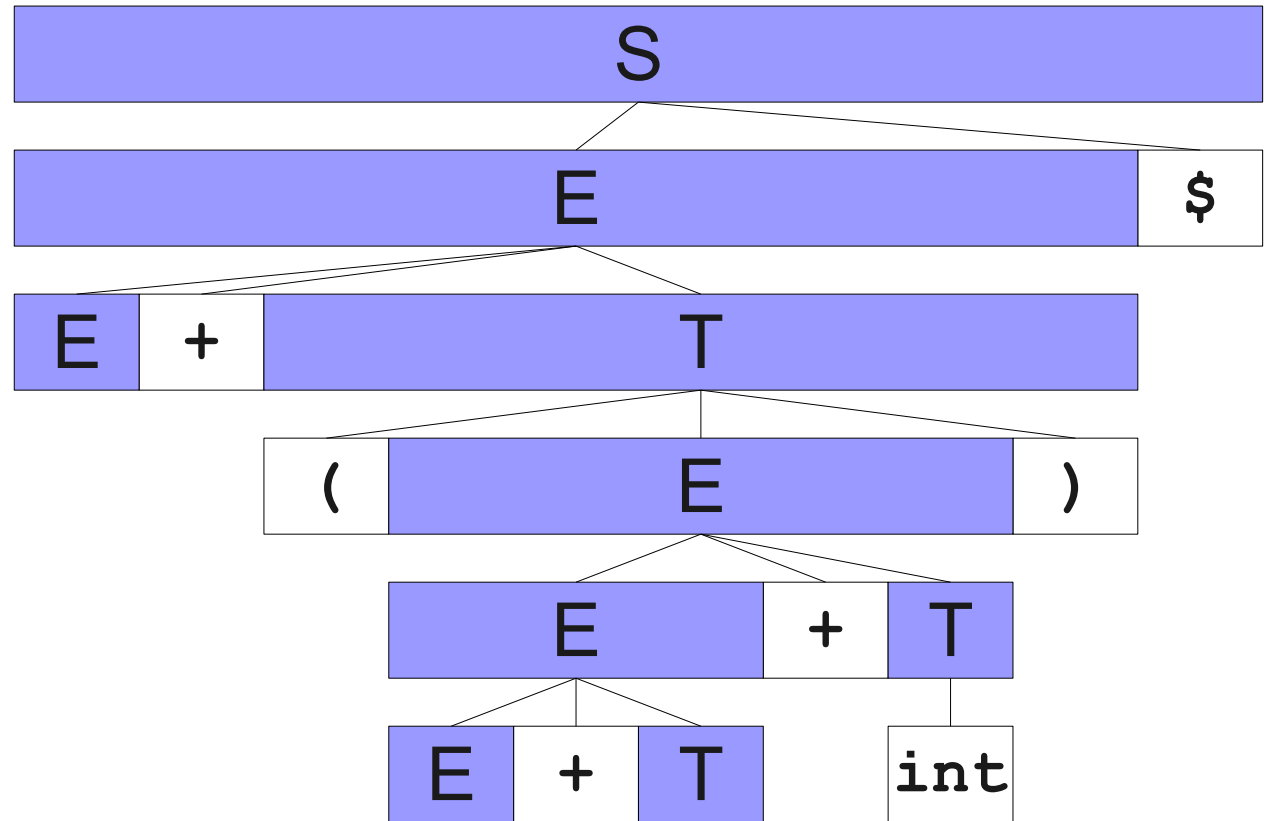
A Third View of a Bottom-Up Parse

`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



A Third View of a Bottom-Up Parse

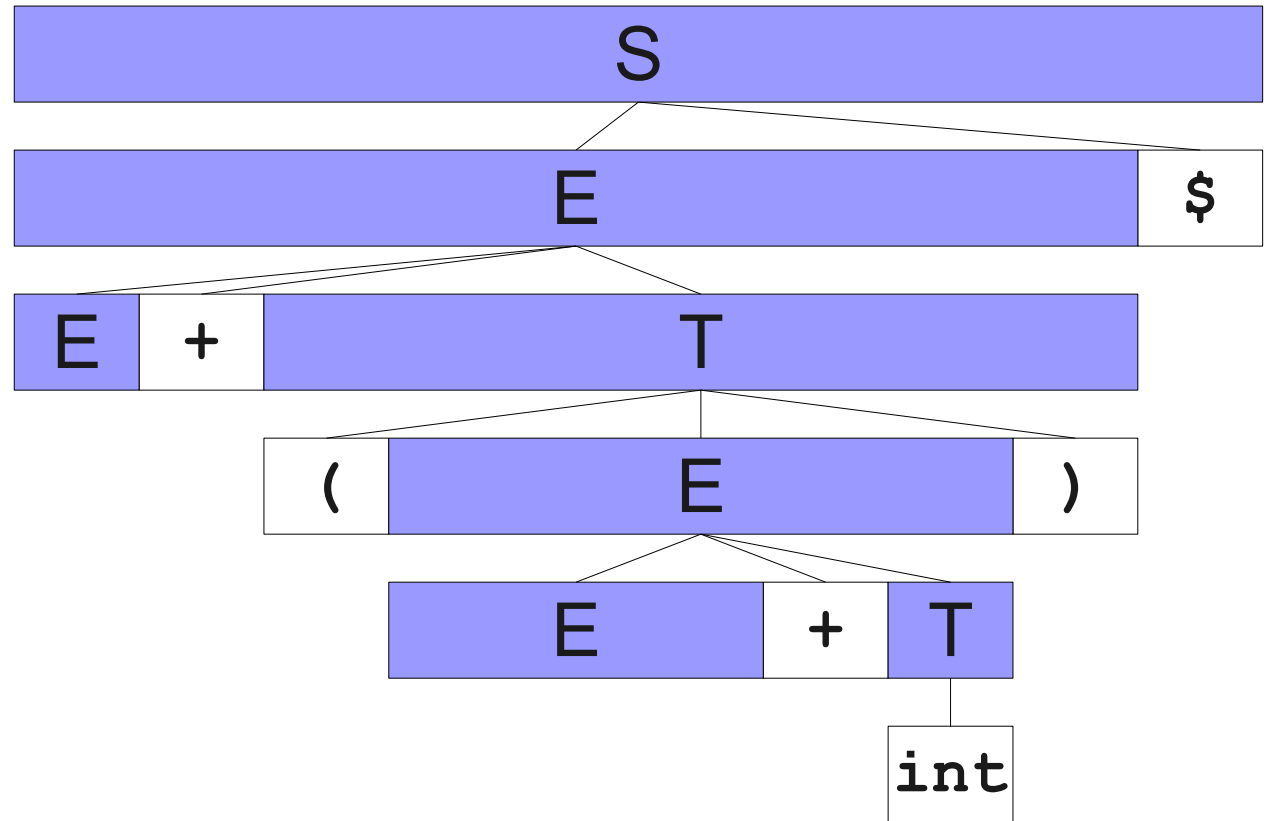
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



`int + (int + int + int) $`

A Third View of a Bottom-Up Parse

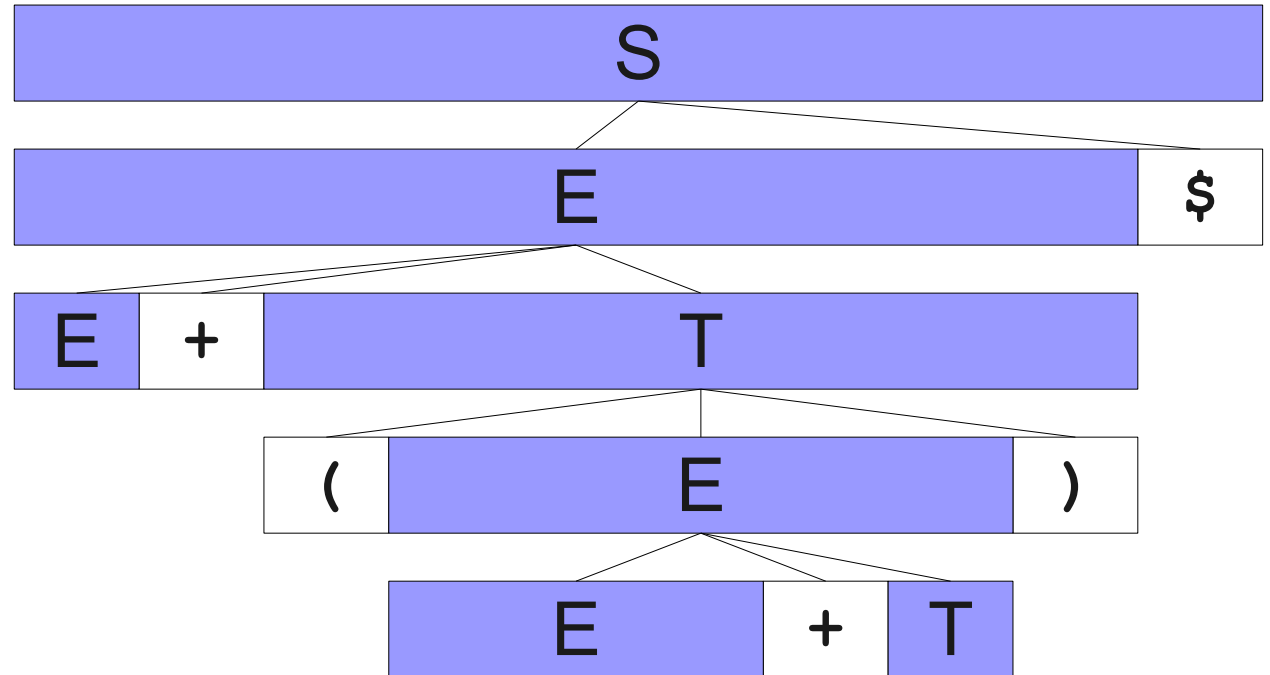
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

A Third View of a Bottom-Up Parse

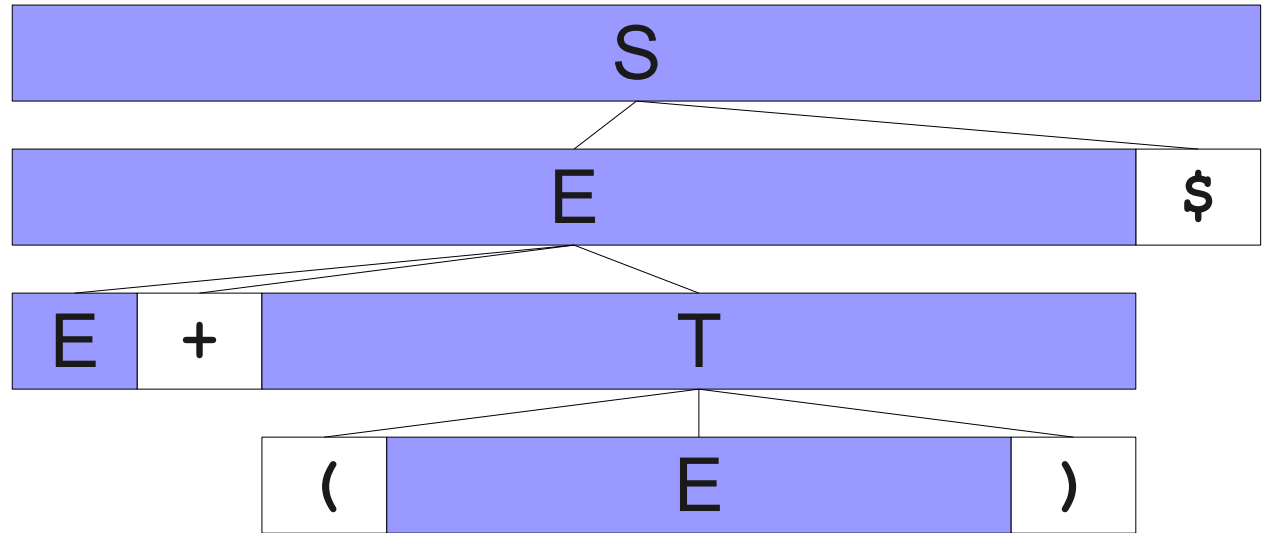
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

A Third View of a Bottom-Up Parse

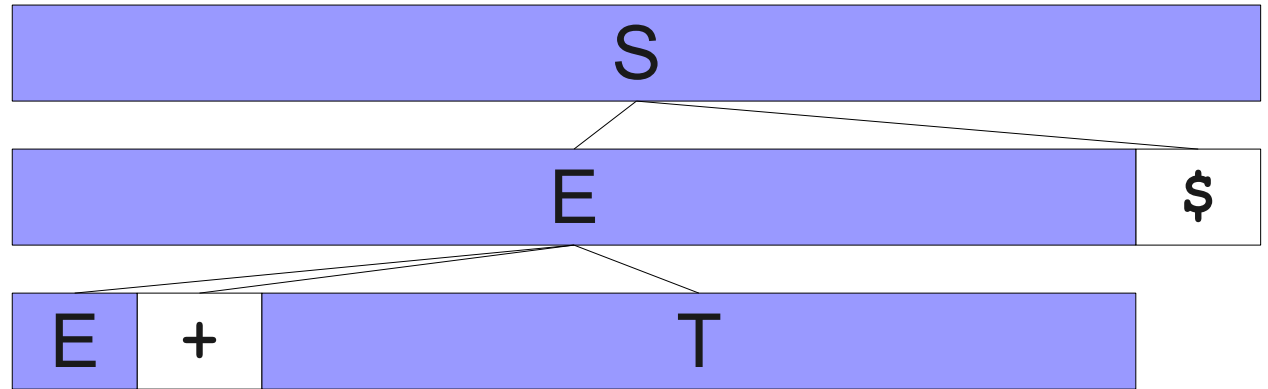
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

A Third View of a Bottom-Up Parse

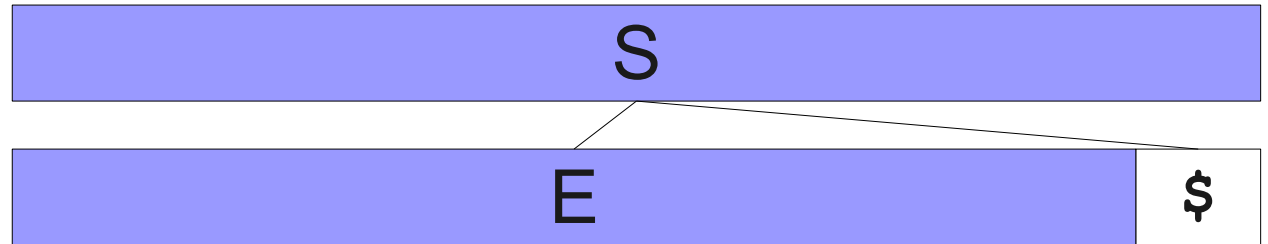
`int + (int + int + int) $`
`T + (int + int + int) $`
`E + (int + int + int) $`
`E + (T + int + int) $`
`E + (E + int + int) $`
`E + (E + T + int) $`
`E + (E + int) $`
`E + (E + T) $`
`E + (E) $`
`E + T $`
`E $`
`S`



<code>int</code>	<code>+</code>	<code>(</code>	<code>int</code>	<code>+</code>	<code>int</code>	<code>+</code>	<code>int</code>	<code>)</code>	<code>\$</code>
------------------	----------------	----------------	------------------	----------------	------------------	----------------	------------------	----------------	-----------------

A Third View of a Bottom-Up Parse

int + (int + int + int) \$
T + (int + int + int) \$
E + (int + int + int) \$
E + (T + int + int) \$
E + (E + int + int) \$
E + (E + T + int) \$
E + (E + int) \$
E + (E + T) \$
E + (E) \$
E + T \$
E \$
S



int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

A Third View of a Bottom-Up Parse

int + (int + int + int) \$
T + (int + int + int) \$
E + (int + int + int) \$
E + (T + int + int) \$
E + (E + int + int) \$
E + (E + T + int) \$
E + (E + int) \$
E + (E + T) \$
E + (E) \$
E + T \$
E \$
S

S

int	+	(int	+	int	+	int)	\$
-----	---	---	-----	---	-----	---	-----	---	----

Handles

- A **reduction** transforms uwv to uAv if $A \rightarrow w$ is a production.
- A **handle** of a sentential form x is a substring w and a production $A \rightarrow w$ where reducing.

$$uwv \rightarrow uAv$$

allows the start symbol to be reached from uAv .

- Informally, a production we can reverse without getting stuck.
- A left-to-right, bottom-up parse **always prunes the leftmost handle**.

A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int	+	int	*	int	\$
-----	---	-----	---	-----	----

A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

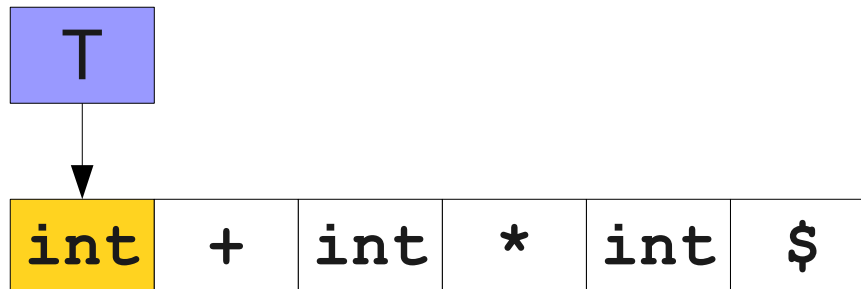
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

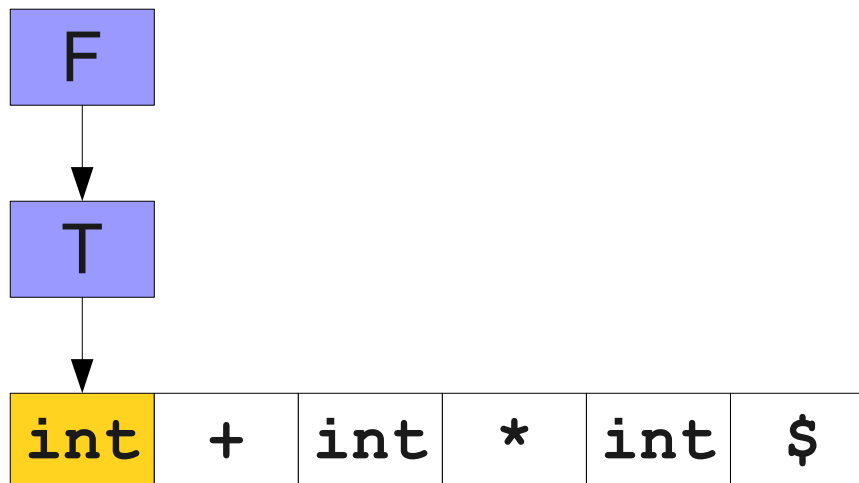
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

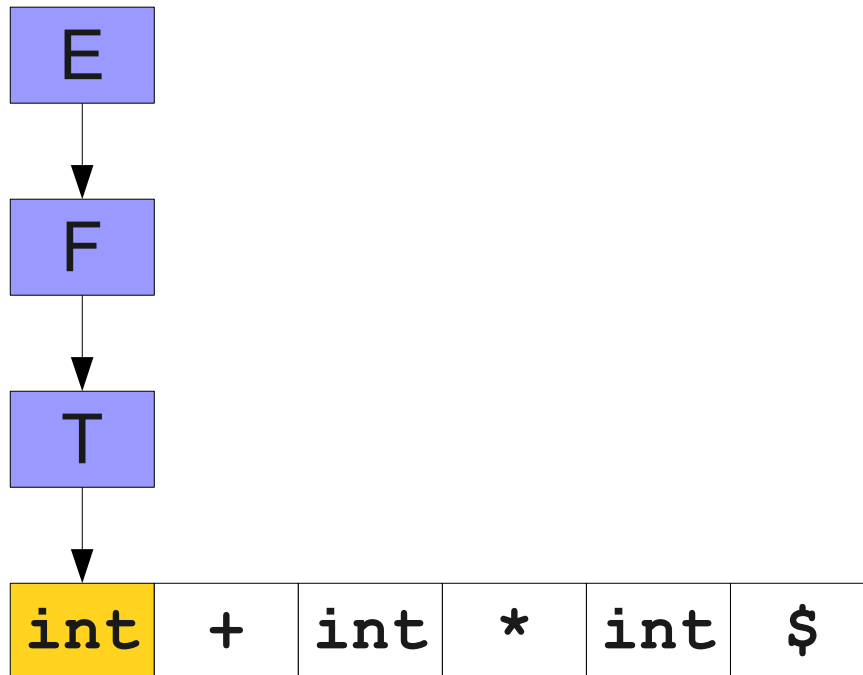
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

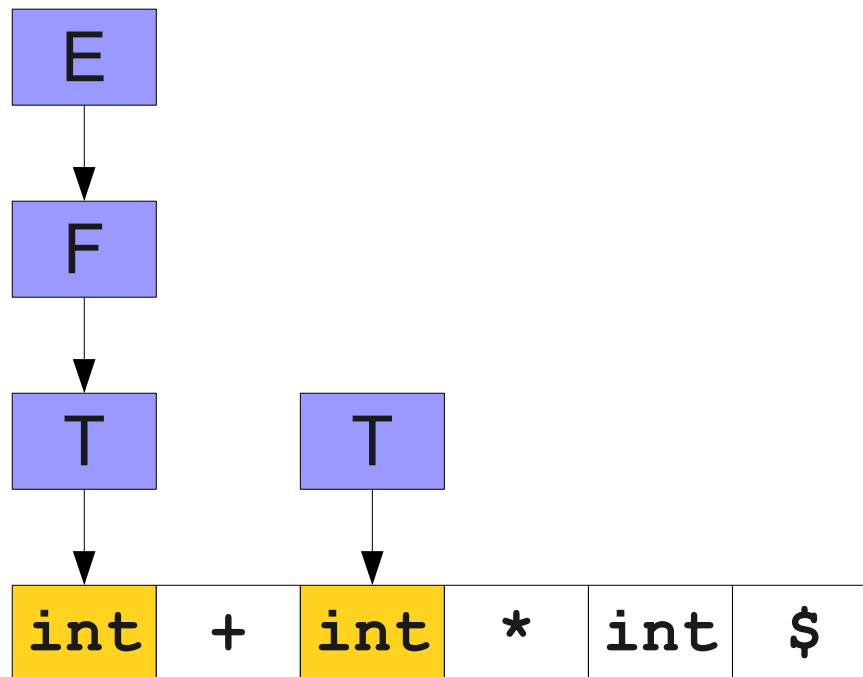
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

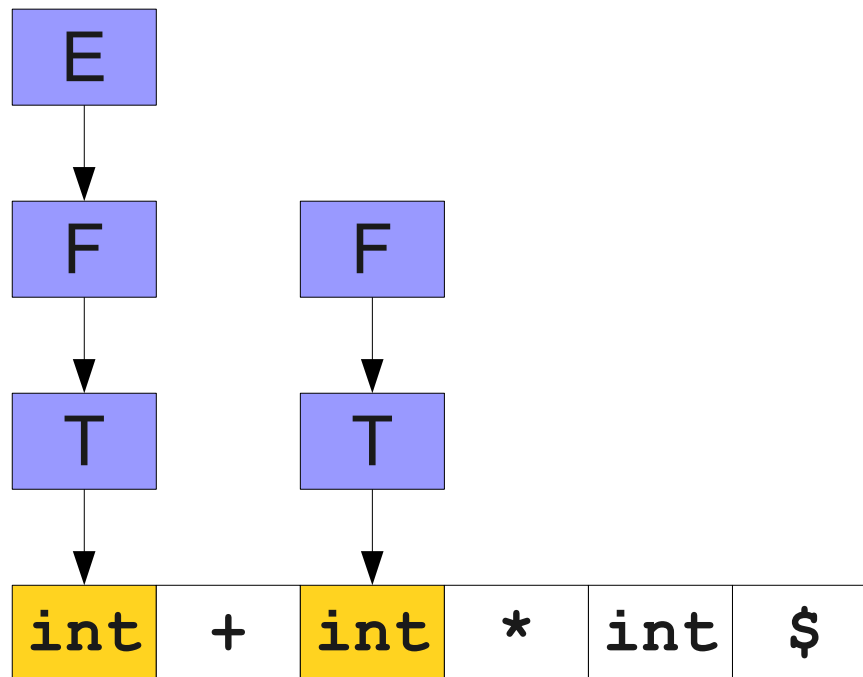
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

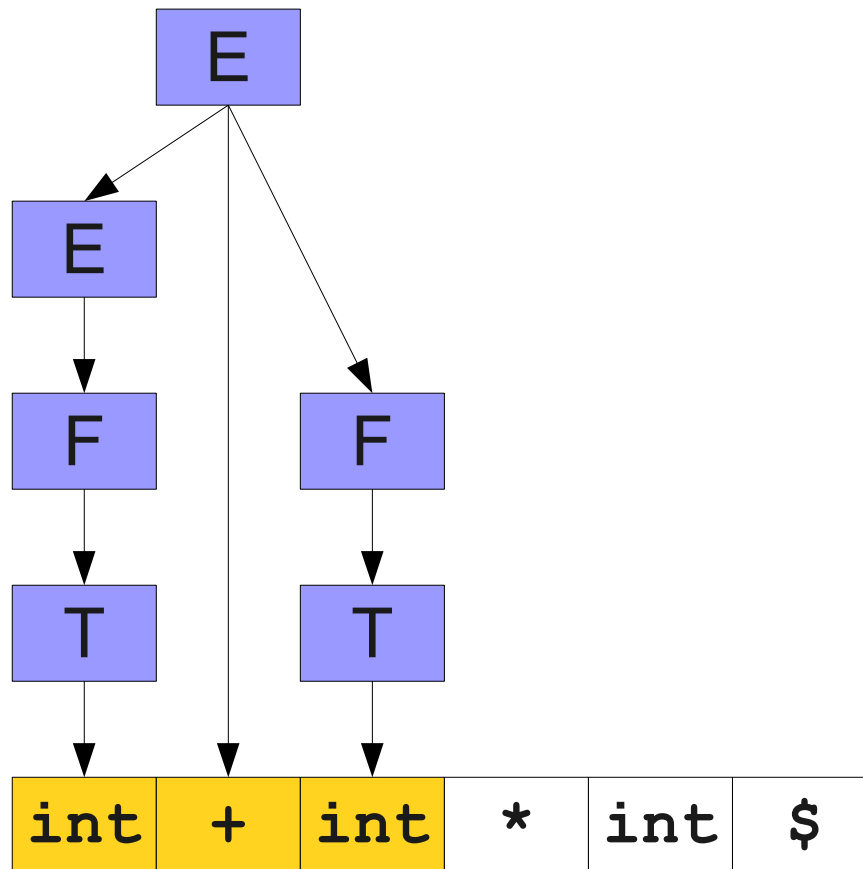
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

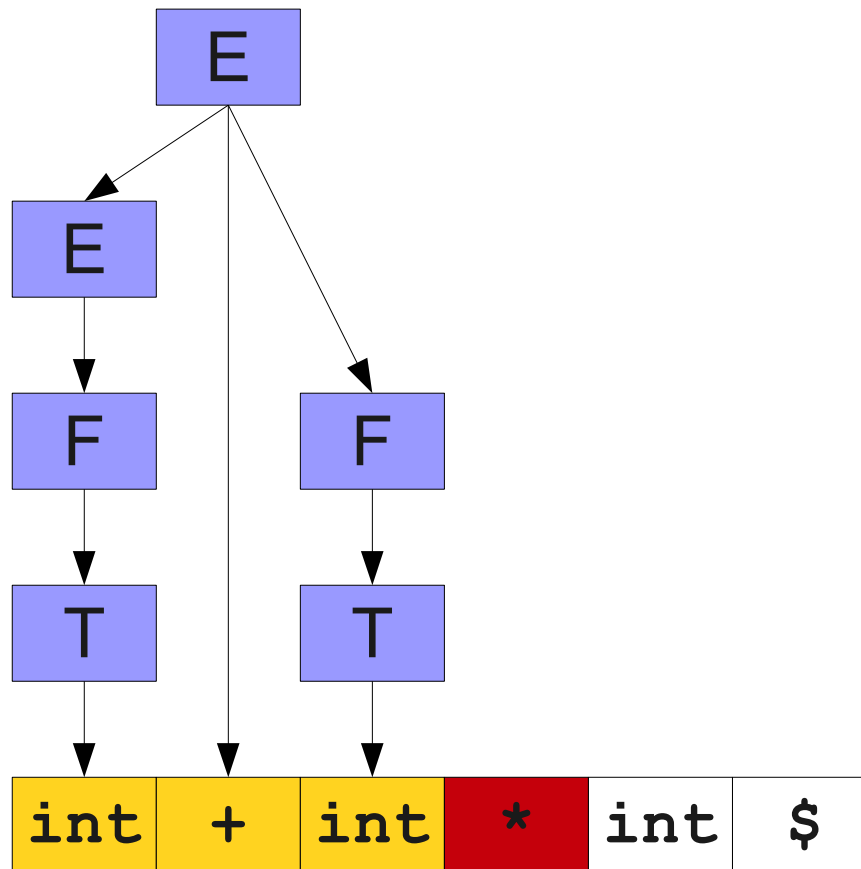
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Detail about Handles

$S \rightarrow E\$$

$E \rightarrow F$

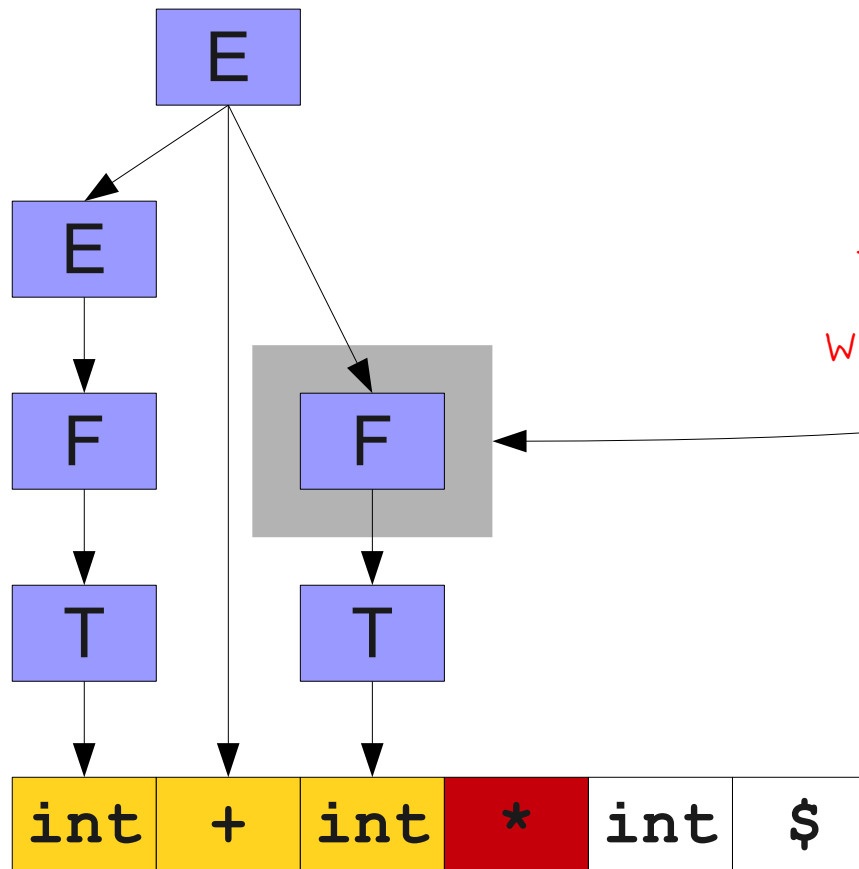
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

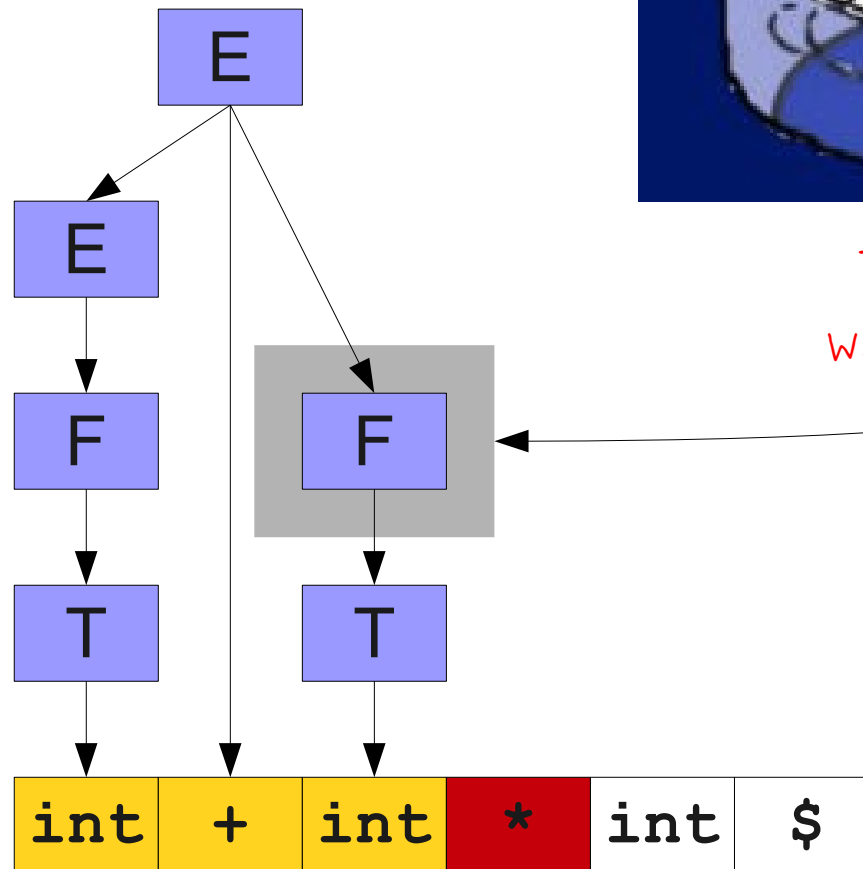


This reduction
wasn't a handle!

A Detail about Handles

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

I LIED!



This reduction wasn't a handle!

The leftmost **reduction** isn't
always the leftmost **handle**.

Finding Handles

- Where do we look for handles?
 - What parts of the string might have a handle in it?
- How do we search for handles?
 - What algorithm do we use to try to discover a handle?
- How do we recognize handles?
 - Once we've found a possible handle, how do we confirm that it's correct?
- **The first two questions have definitive answers.**

Question One:

Where are handles?



Right here...



Where can we find handles?

- Recall: left-to-right, bottom-up parse is a **rightmost derivation traced in reverse**.
- Suppose that our current form is uvw and the handle is $A \rightarrow v$.
- Then after reducing the handle, we get uAw .
- Can w contain any nonterminals?
- **No**: Otherwise we would have reduced a handle somewhere in w .

Shift/Reduce Parsing

- Idea: Split the input into two parts:
 - **Left** substring is our work area.
 - **Right** substring is input we have not yet processed.
- All handles are reduced in the left substring.
- Right substring consists only of terminals.
- At each point, decide whether to:
 - Move a terminal across the split (**shift**)
 - Reduce a handle (**reduce**)

A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$


$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



int	+	int	*	int	+	int	\$
-----	---	-----	---	-----	---	-----	----

A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

int

int

+

int

*

int

+

int

\$

A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

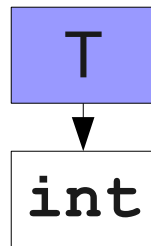
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

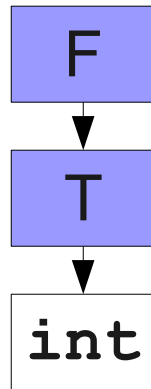
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

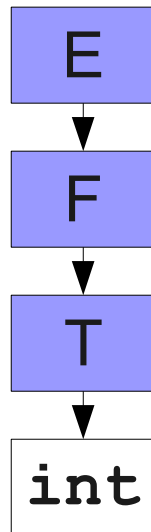
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

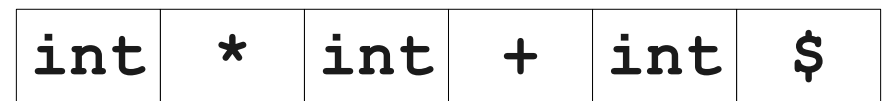
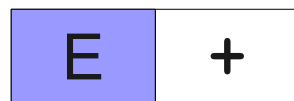
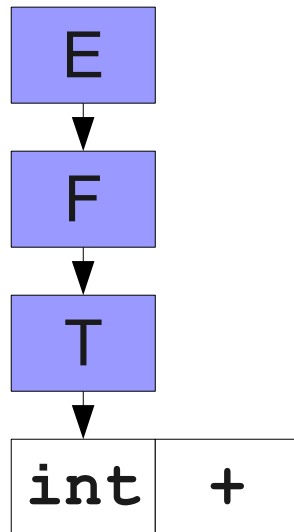
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

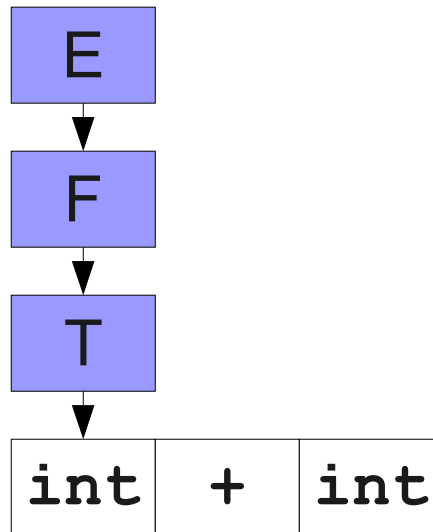
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

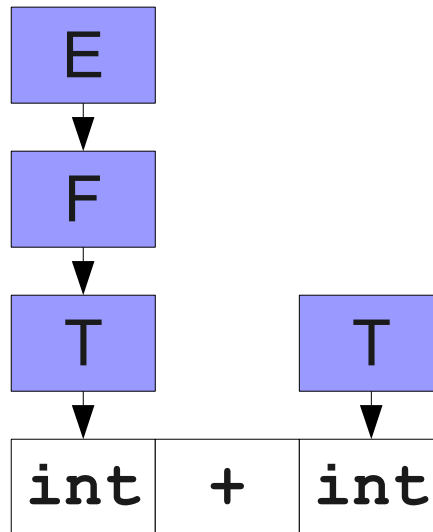
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

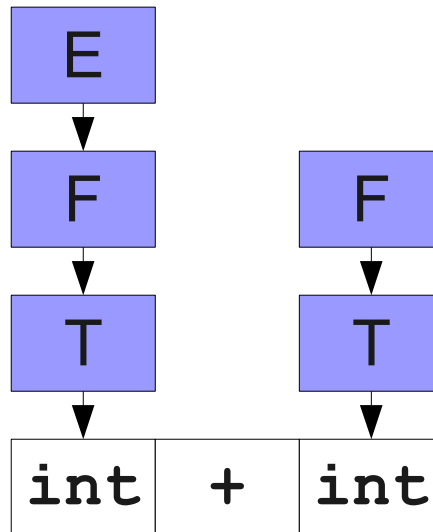
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

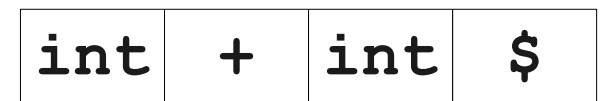
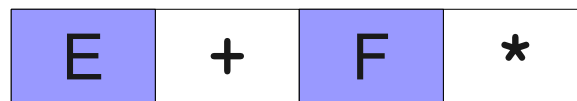
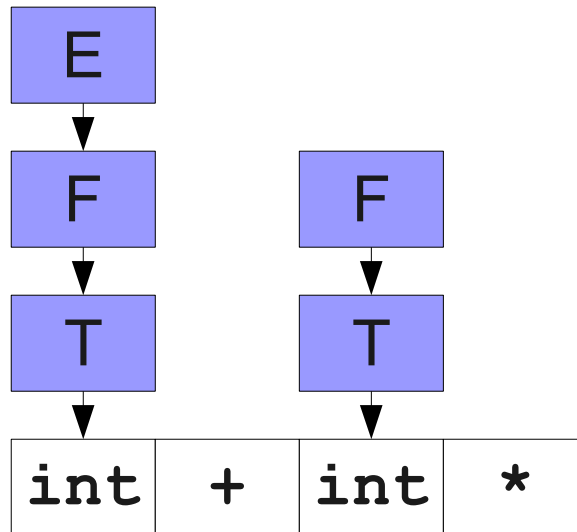
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

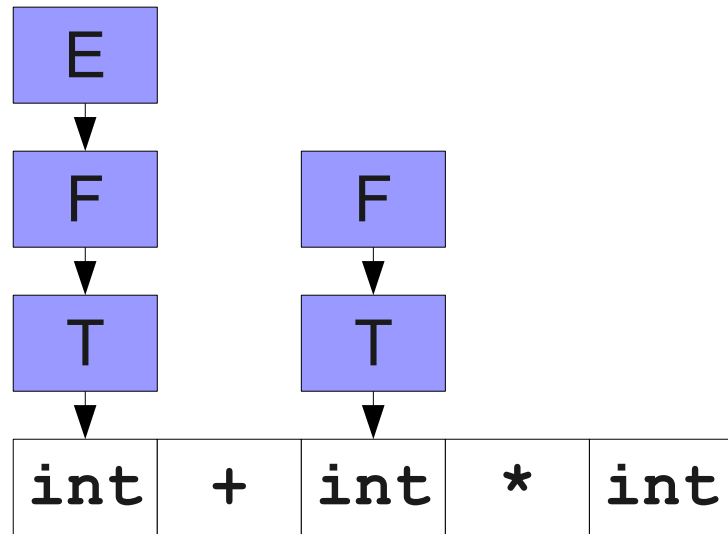
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

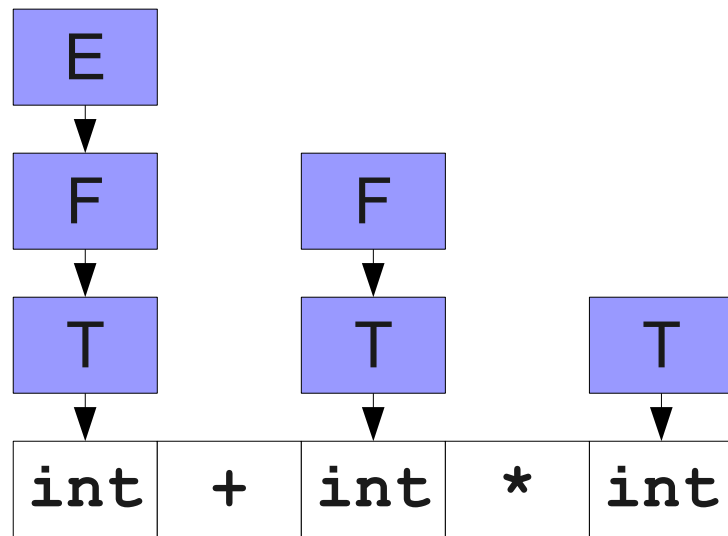
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$

$E \rightarrow F$

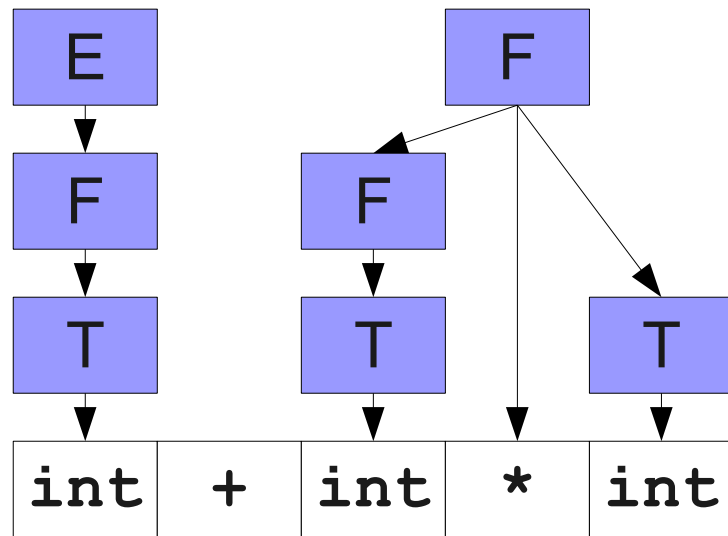
$E \rightarrow E + F$

$F \rightarrow F * T$

$F \rightarrow T$

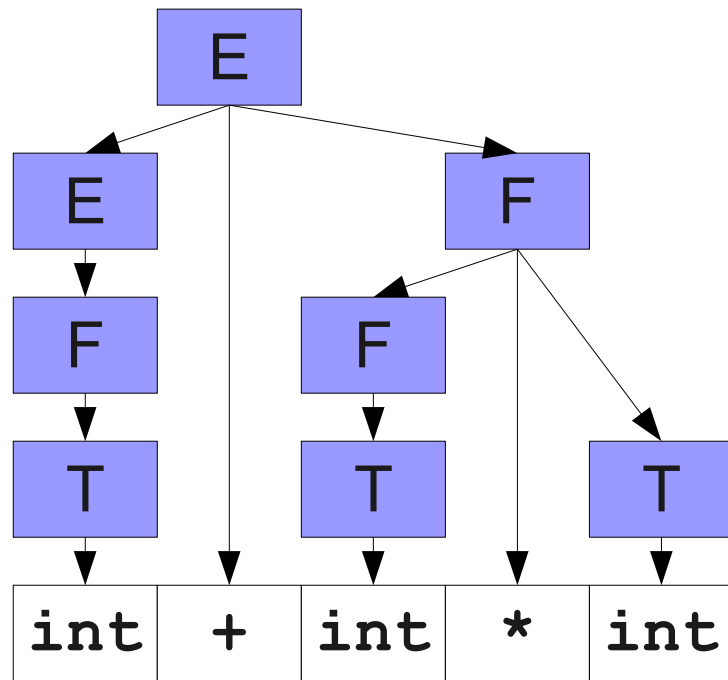
$T \rightarrow \text{int}$

$T \rightarrow (E)$



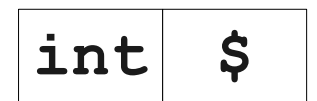
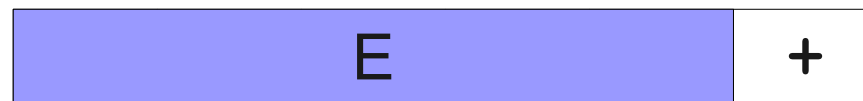
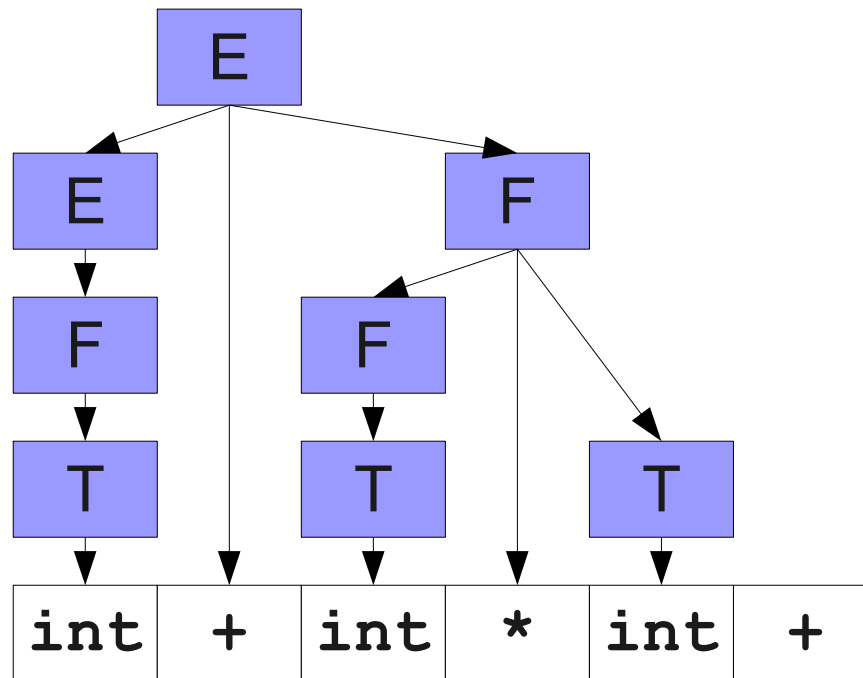
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



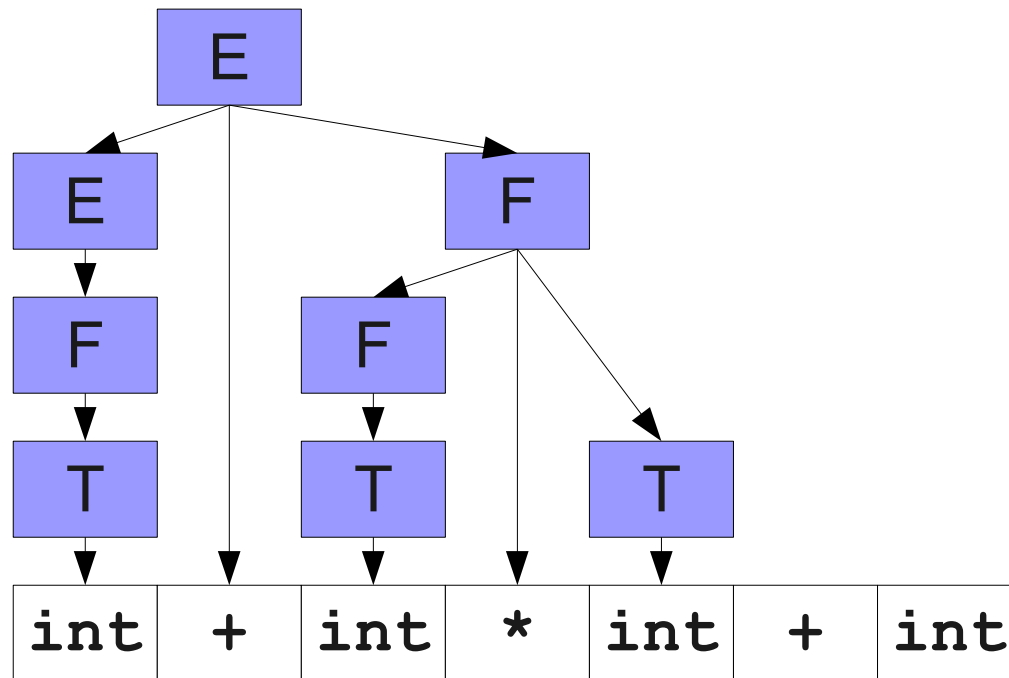
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



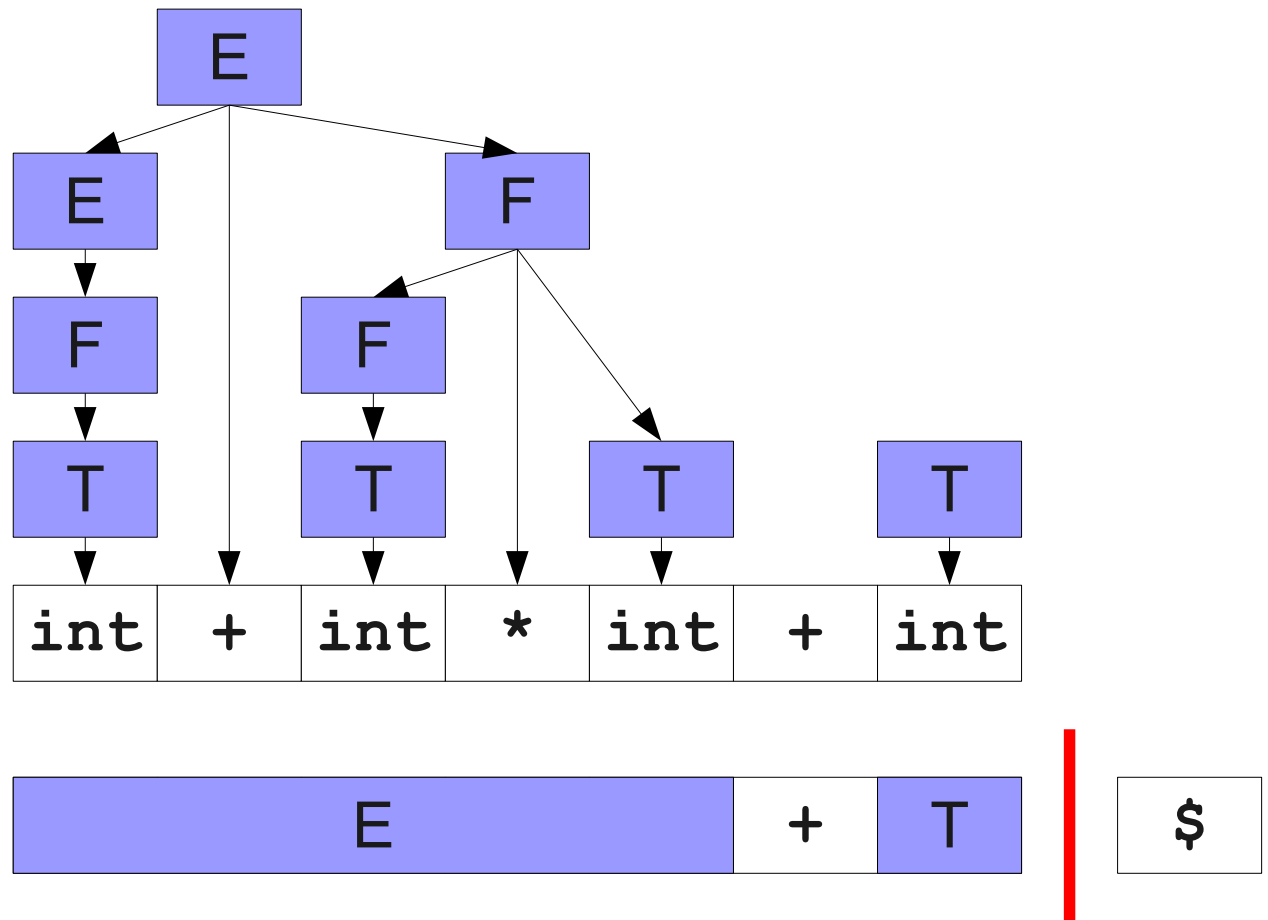
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



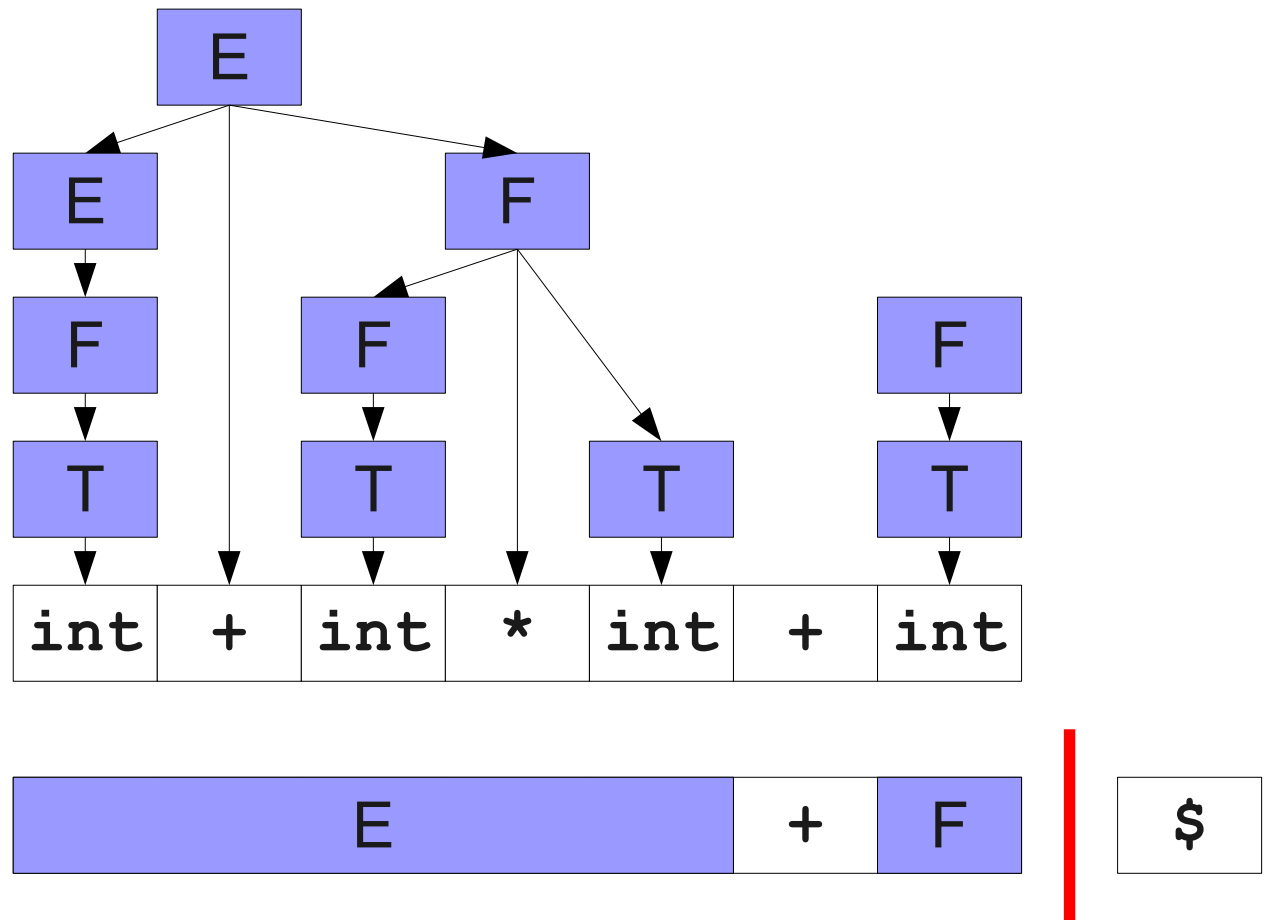
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



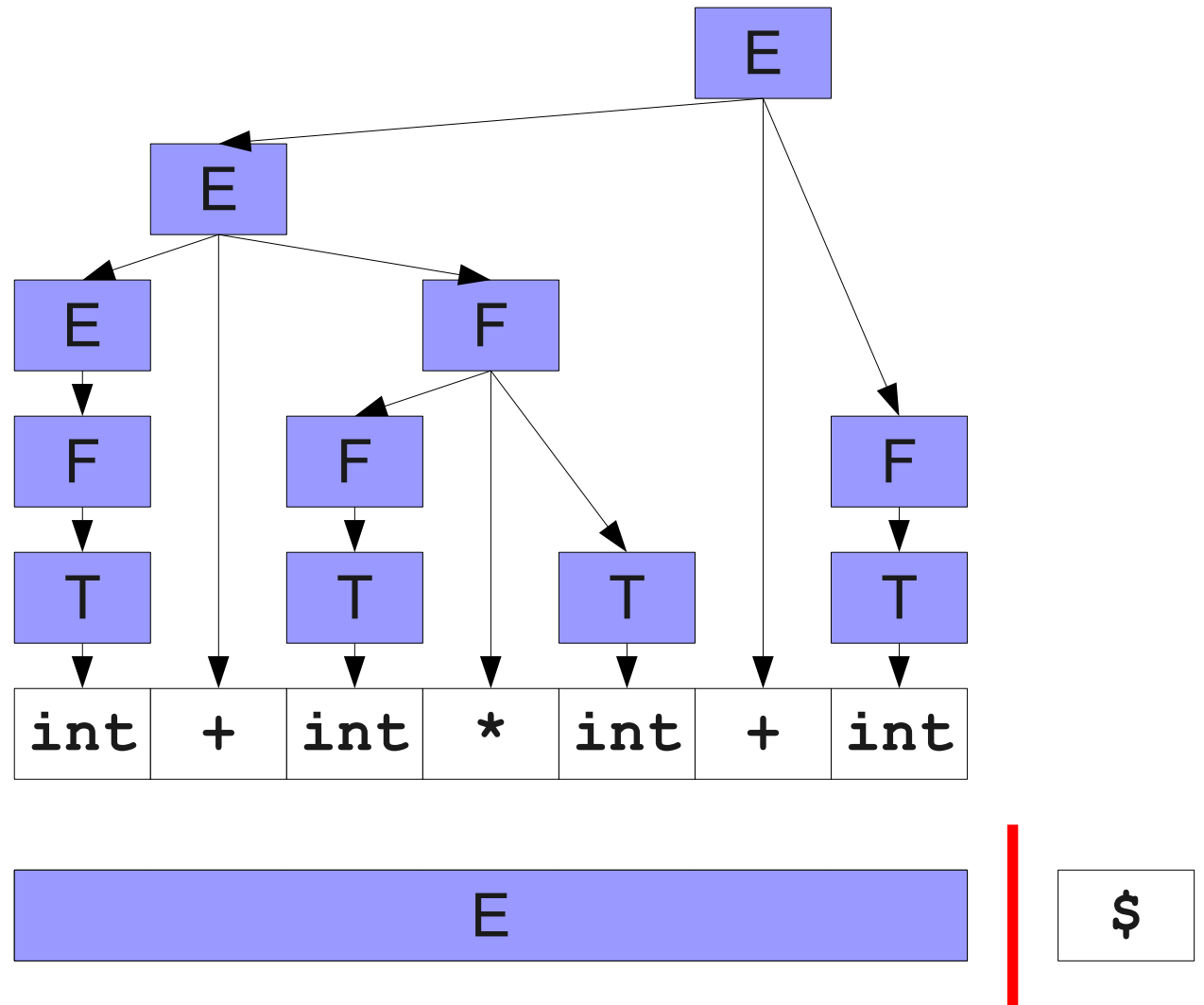
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



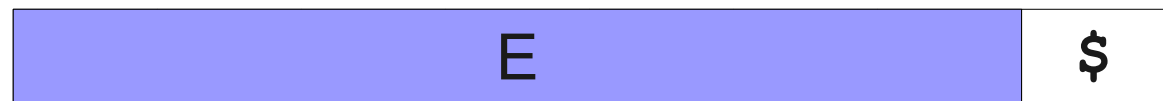
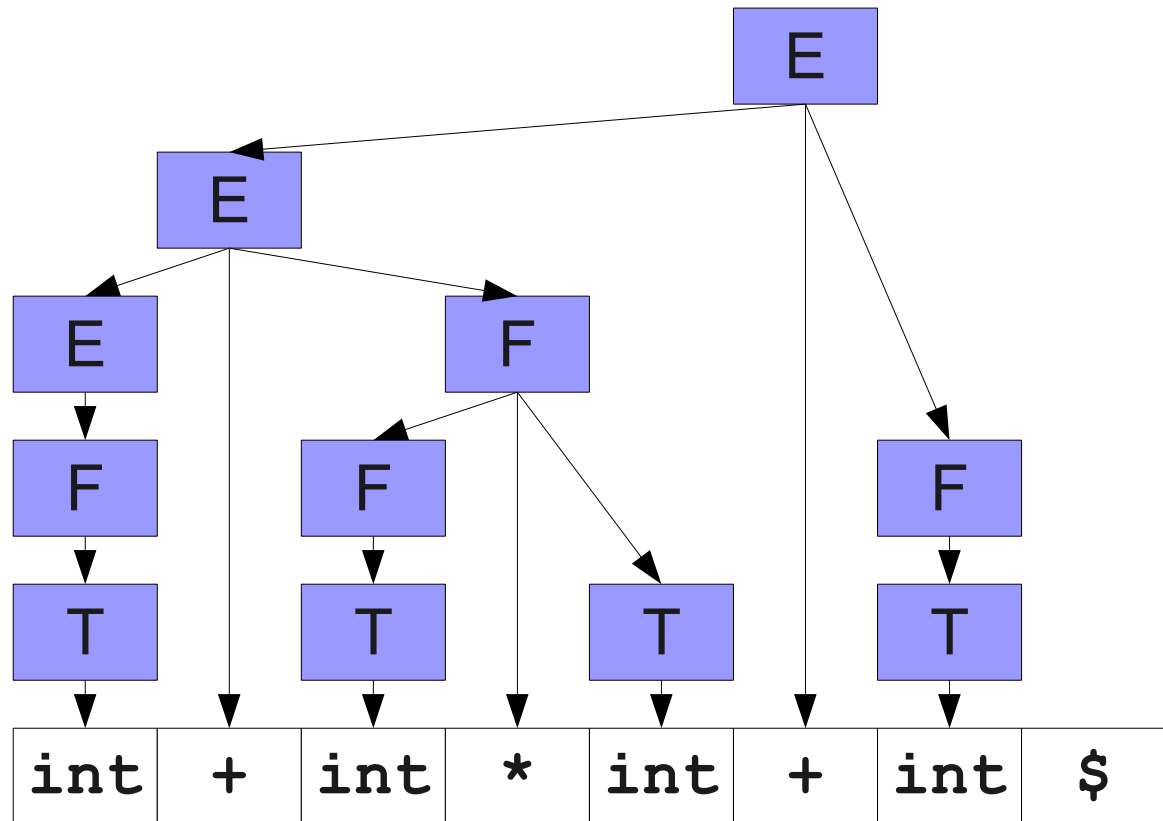
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



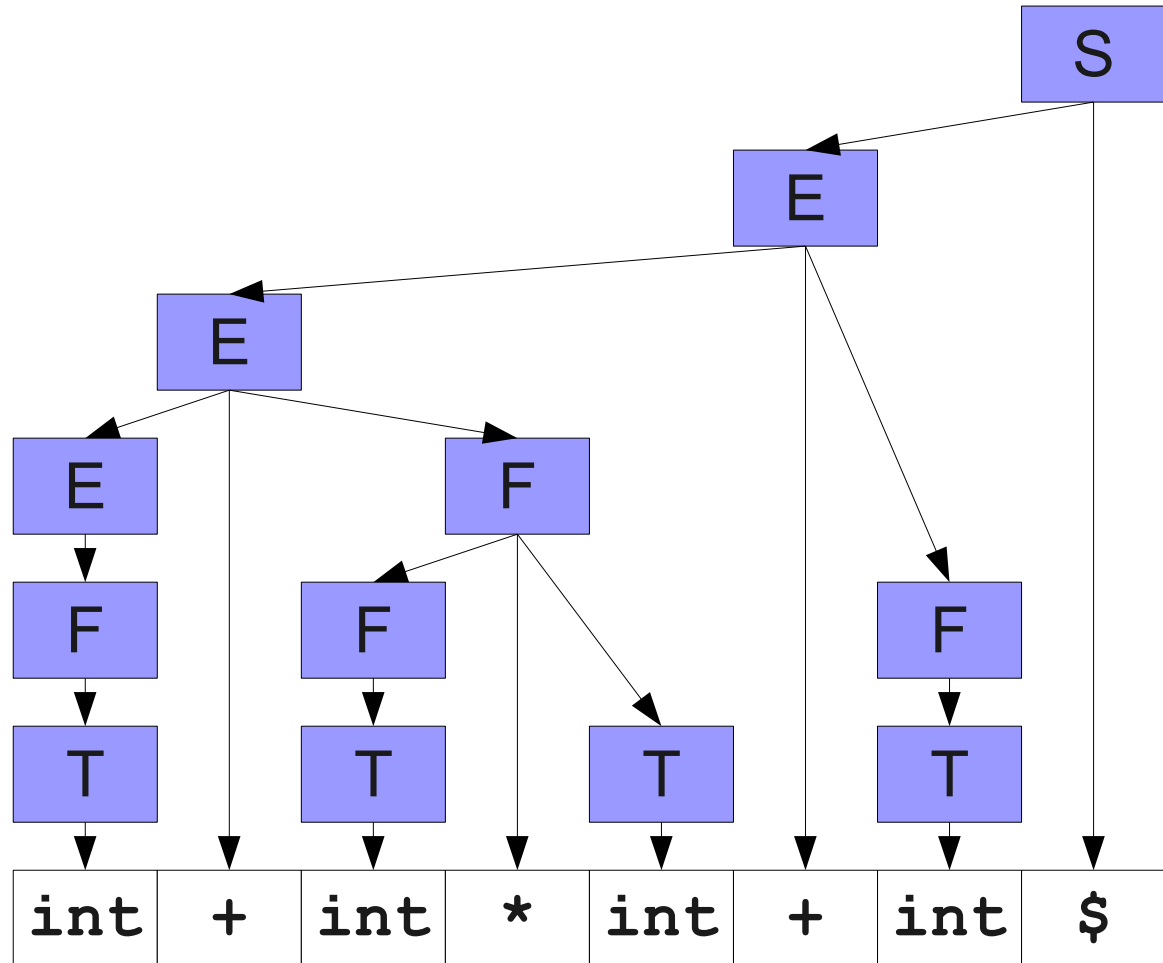
A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



A Sample Shift/Reduce Parse

$S \rightarrow E\$$
 $E \rightarrow F$
 $E \rightarrow E + F$
 $F \rightarrow F * T$
 $F \rightarrow T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$



S



An Important Observation

- All of the reductions we applied were to the far right end of the left area.
- **This is not a coincidence!**
- No matter what grammar we use, this will be true.
- Proof sketch: by induction on the number of reduces
 - After no reduces, the first reduction can be done at the right end of the left area.
 - After at least one reduce, the very right of the left area is a nonterminal. This nonterminal must be part of the next reduction, since we're tracing a rightmost derivation backwards.

An Important Corollary

- Since reductions are always at the right side of the left area, we **never need to shift from the left to the right.**
- Why?
 - No need to “uncover” something to do a reduction.
- Consequently, shift/reduce parsing means
 - **Shift:** Move a terminal from the right to the left area.
 - **Reduce:** Replace some number of symbols at the right side of the left area.

Simplifying our Terminology

- All activity in a shift/reduce parser is at the far right end of the left area.
- Idea: Represent the left area as a **stack**.
- **Shift**: Push the next terminal onto the stack.
- **Reduce**: Pop some number of symbols from the stack, then push the appropriate nonterminal.

Finding Handles

- Where do we look for handles?
 - **At the top of the stack.**
- How do we search for handles?
 - What algorithm do we use to try to discover a handle?
- How do we recognize handles?
 - Once we've found a possible handle, how do we confirm that it's correct?

Question Two:

How do we search for handles?

Searching for Handles

- When using a shift/reduce parser, we must decide whether to shift or reduce at each point.
- We only want to reduce when we know we have a handle.
- **Question:** How can we tell that we might be looking at a handle?

Viable Prefixes

- A **viable prefix** is a string w such that, for some v , $w \mid v$ is a legal configuration in a shift/reduce parser.
- In other words, a viable prefix is
a string of terminals and nonterminals,
that does not extend past a handle,
that can be completed with terminals into a
sentential form.

Why Care about Viable Prefixes?

- As long as the stack contains a viable prefix, we can always extend what we have.
- Leads to a naïve parsing algorithm:
 - **Shift** if doing so leaves the stack containing a viable prefix.
 - Otherwise, **reduce** if we have found a handle and **error** if the input is malformed.

Critical Fact

The set of viable prefixes for any grammar is a **regular language**.

Critical Fact

The set of viable prefixes for any grammar is a **regular language**.

This is a nontrivial result!

Viable Prefixes are Regular: Intuition

- Intuition: Unbounded memory makes some context-free languages not regular.
 - For example: balanced parentheses.
- A viable prefix does not contain any complete productions.
 - If it did, one would be a handle and we could reduce it.
- Consequently, we can't **use** any of that memory.

Viabable Prefixes are Regular: Intuition

- A viable prefix is the concatenation of several partial productions.
- Keep track of **which productions** and **where we are** in those productions.
- Finitely many places you can be in a particular production.
- Finitely many different productions to choose at each point.
- **Build an automaton** to track the current production and current position.

Tracking our Position in Productions


$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$



int	+	(int	+	int)	\$
-----	---	---	-----	---	-----	---	----

Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$

| int + (int + int) \$

Tracking our Position in Productions

$S \rightarrow E\$$


$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$



<code>int</code>	<code>+</code>	<code>(</code>	<code>int</code>	<code>+</code>	<code>int</code>	<code>)</code>	<code>\$</code>
------------------	----------------	----------------	------------------	----------------	------------------	----------------	-----------------

Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$

| int + (int + int) \$

Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \cdot \text{int}$

| int + (int + int) \$

Tracking our Position in Productions

$S \rightarrow E\$$

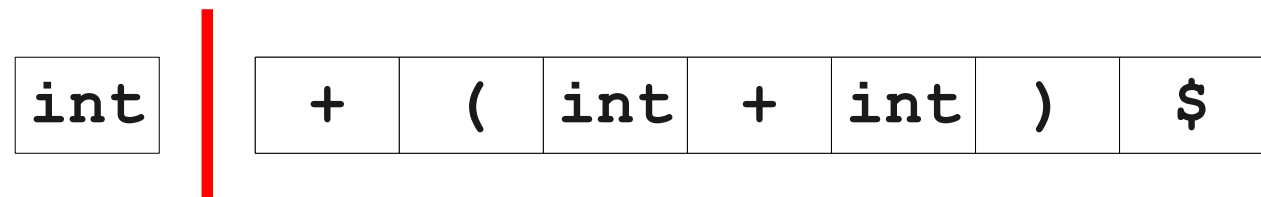
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \cdot \text{int}$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

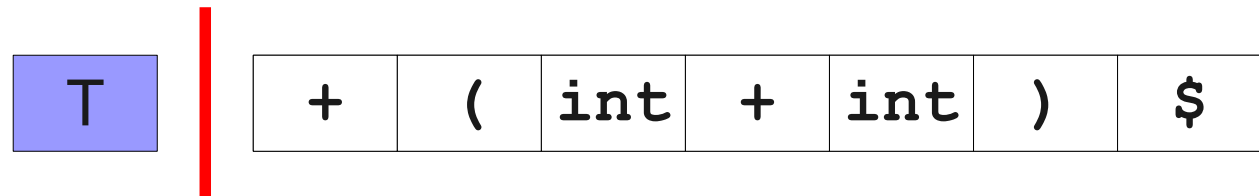
$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \text{int} \cdot$

int		+	(int	+	int)	\$
-----	--	---	---	-----	---	-----	---	----

Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \text{int} \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow T \cdot$

T

+	(int	+	int)	\$
---	---	-----	---	-----	---	----

Tracking our Position in Productions

$S \rightarrow E\$$

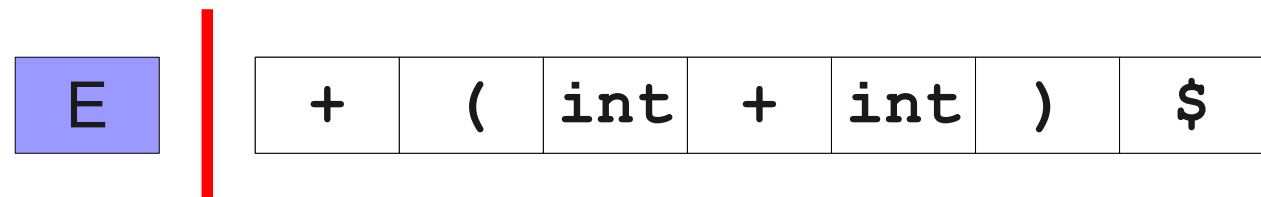
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow \cdot E + T$
$E \rightarrow T \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

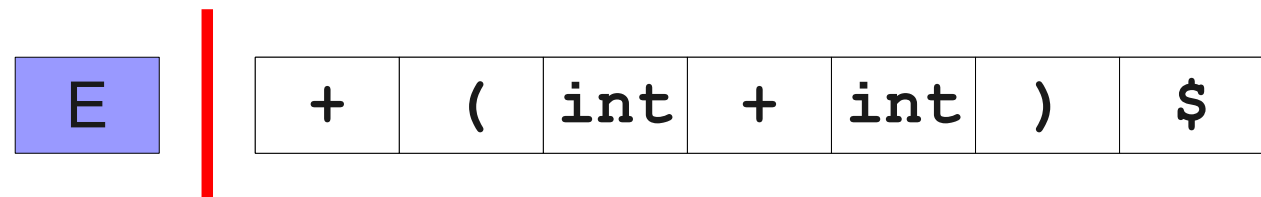
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E \cdot + T$



Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E \cdot + T$

E	+
---	---

(int	+	int)	\$
---	-----	---	-----	---	----

Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$

E	+
---	---

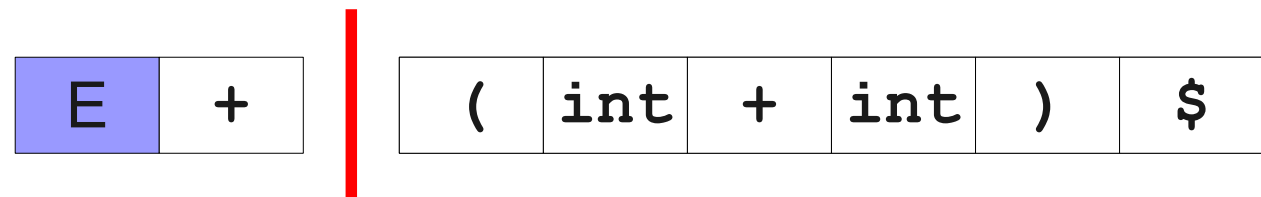


(int	+	int)	\$
---	-----	---	-----	---	----

Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow \cdot (E)$



Tracking our Position in Productions

$S \rightarrow E\$$

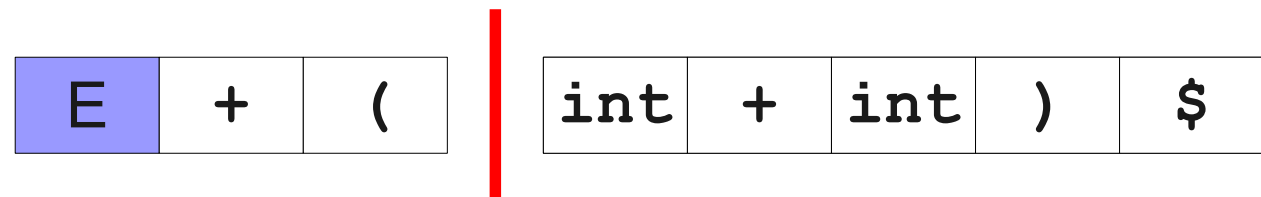
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

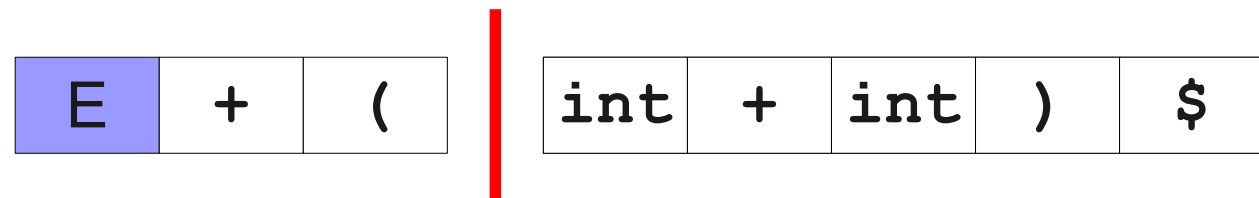
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow \cdot (E)$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$



Tracking our Position in Productions

$S \rightarrow E\$$

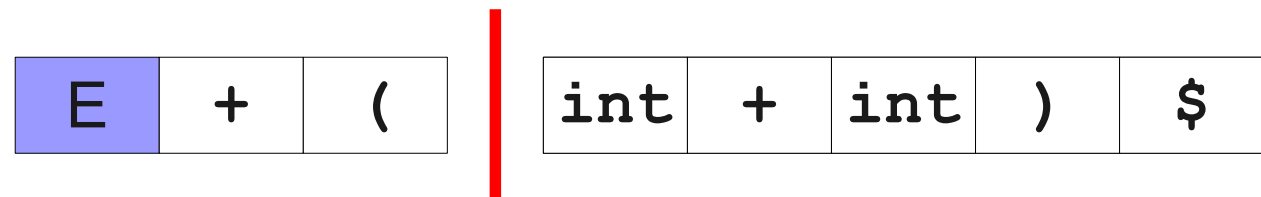
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$



Tracking our Position in Productions

$S \rightarrow E\$$

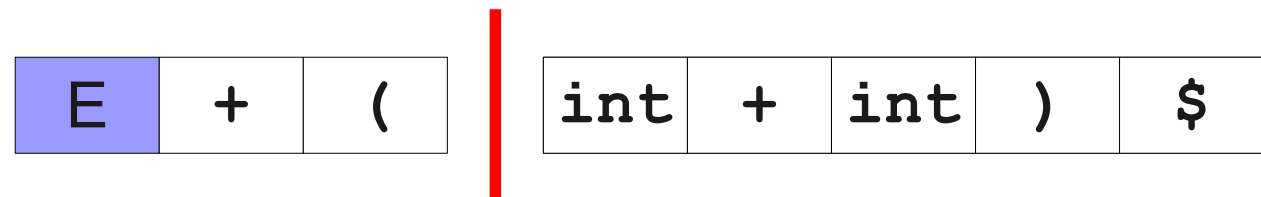
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

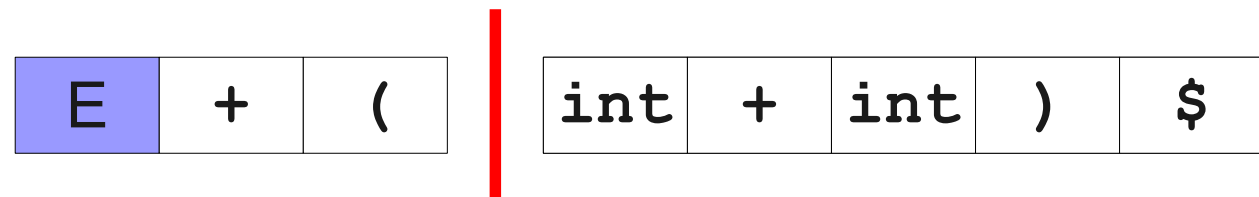
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

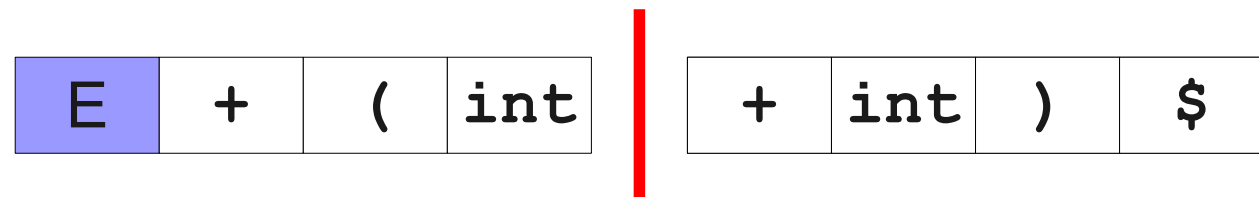
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \cdot \text{int}$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

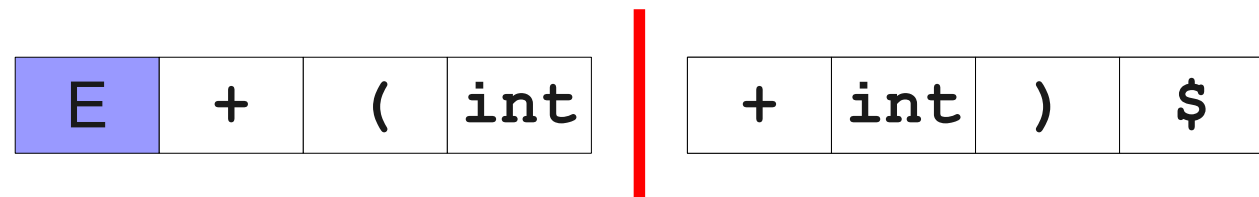
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \cdot \text{int}$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

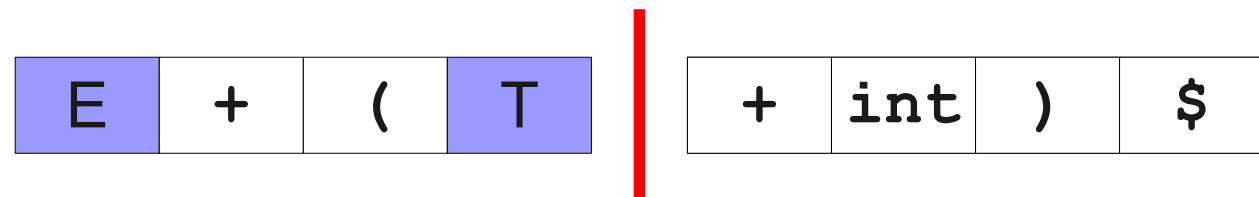
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \text{int} \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

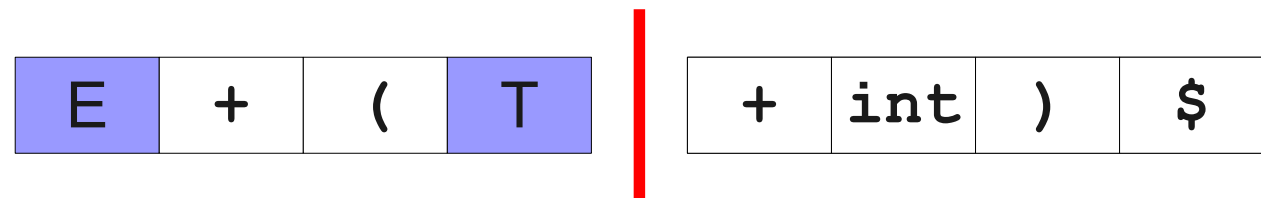
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow \cdot T$
$T \rightarrow \text{int} \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

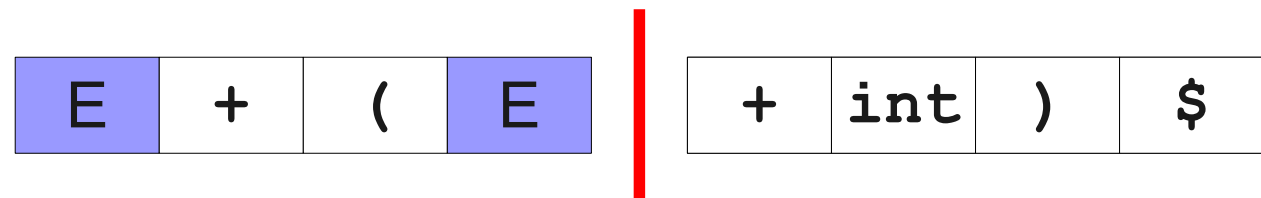
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow T \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

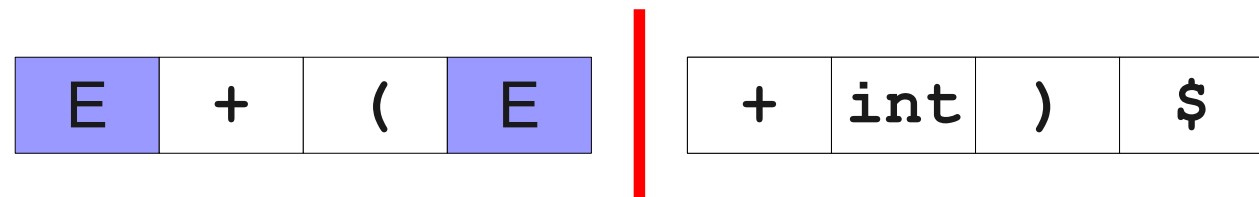
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$
$E \rightarrow T \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow \cdot E + T$



Tracking our Position in Productions

$S \rightarrow E\$$

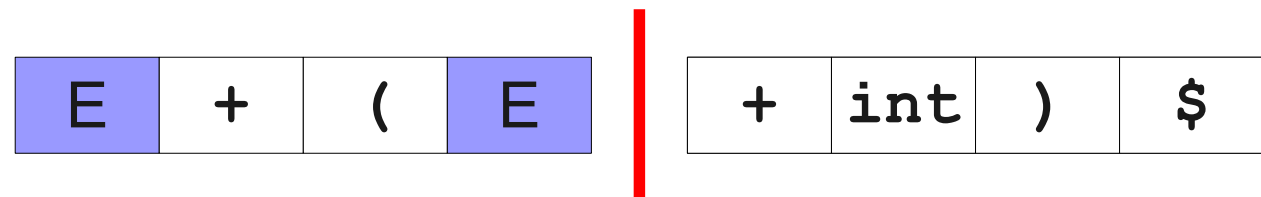
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E \cdot + T$



Tracking our Position in Productions

$S \rightarrow E\$$

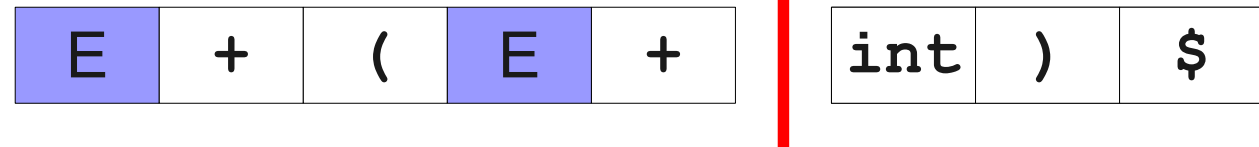
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

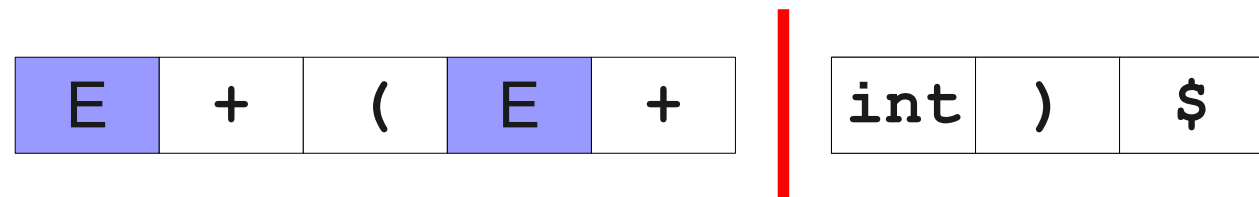
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E \cdot + T$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

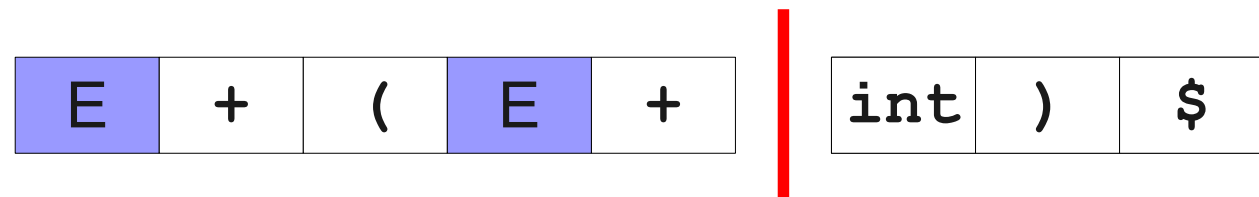
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + \cdot T$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + \cdot T$
$T \rightarrow \cdot \text{int}$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

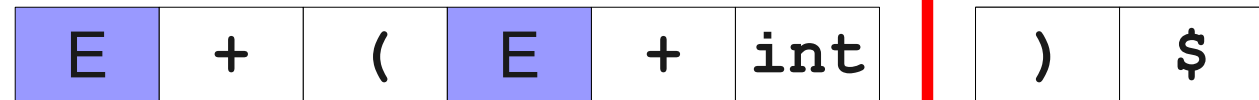
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + \cdot T$
$T \rightarrow \cdot \text{int}$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + \cdot T$
$T \rightarrow \text{int} \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

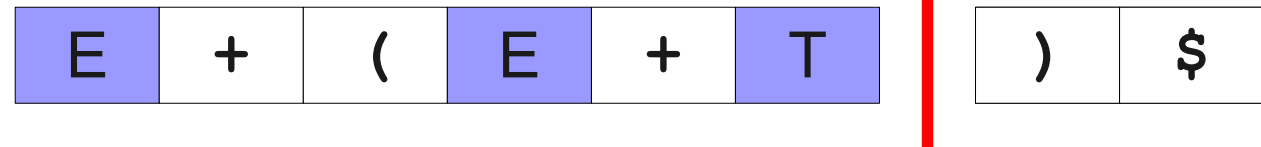
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + \cdot T$
$T \rightarrow \text{int} \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

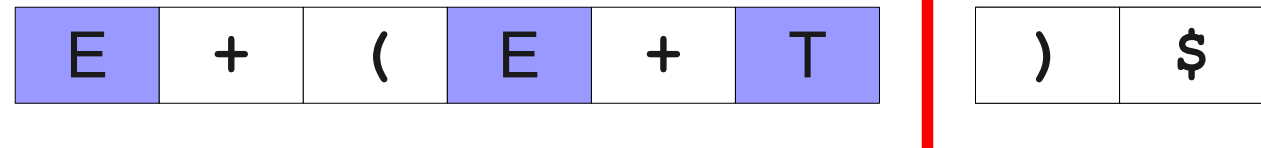
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

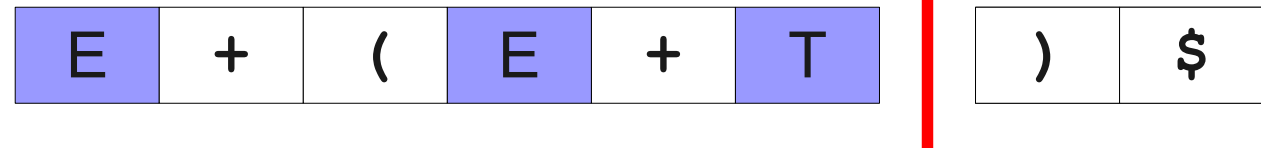
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + \cdot T$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + T \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

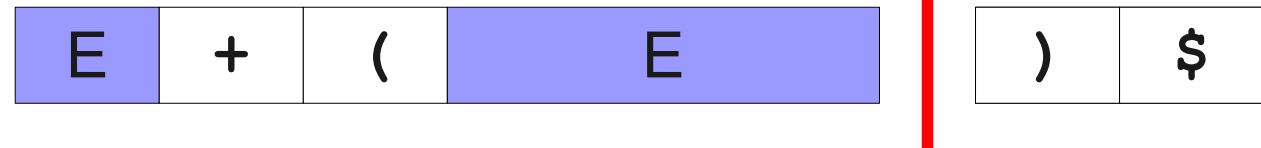
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$
$E \rightarrow E + T \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

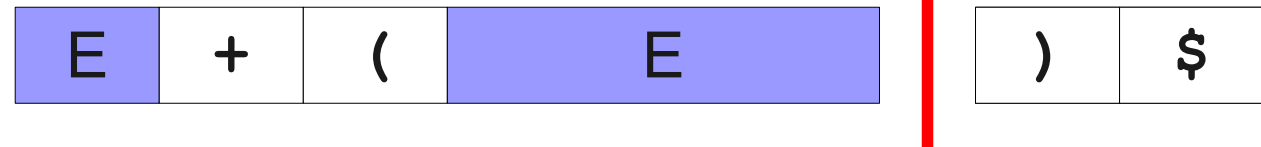
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

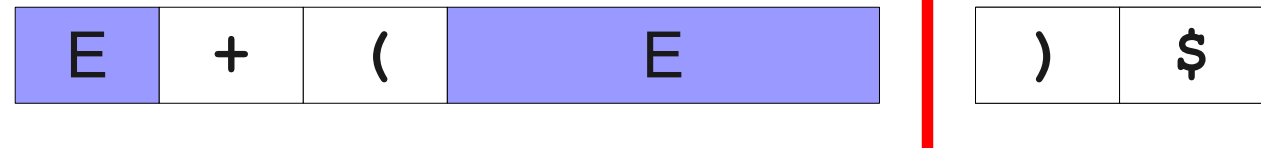
$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (\cdot E)$



Tracking our Position in Productions

$S \rightarrow E\$$
 $E \rightarrow T$
 $E \rightarrow E + T$
 $T \rightarrow \text{int}$
 $T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (E \cdot)$



Tracking our Position in Productions

$S \rightarrow E\$$

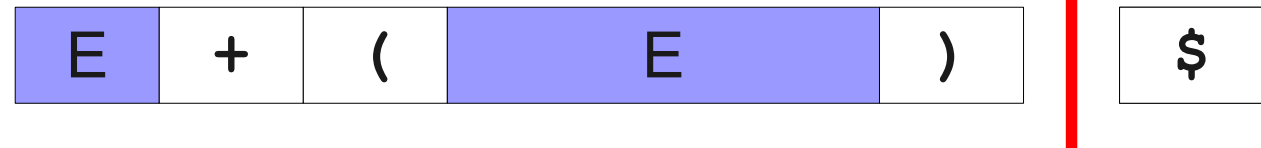
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (E \cdot)$



Tracking our Position in Productions

$S \rightarrow E\$$

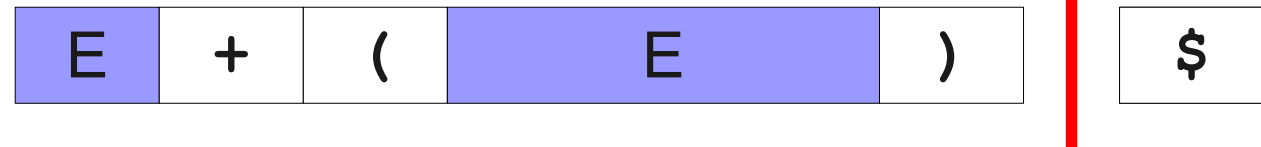
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (E) \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

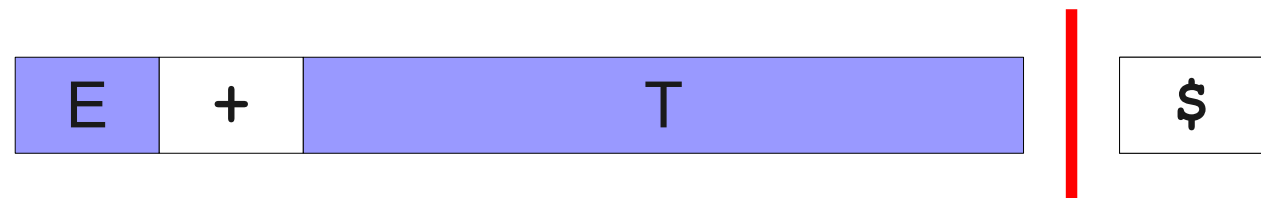
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$
$T \rightarrow (E) \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

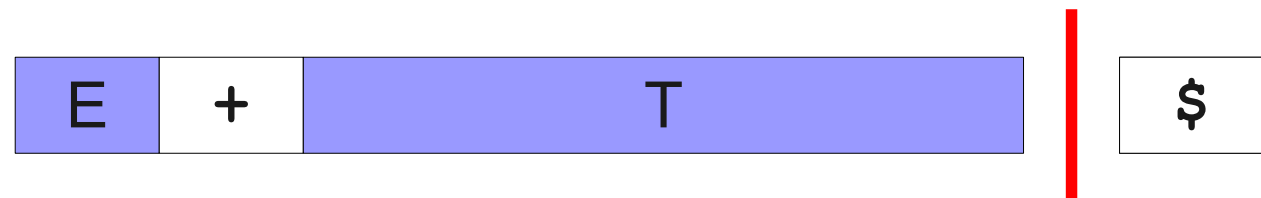
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + \cdot T$



Tracking our Position in Productions

$S \rightarrow E\$$

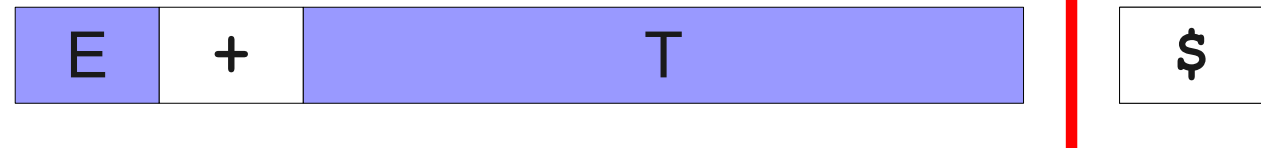
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$
$E \rightarrow E + T \cdot$



Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$

$E \rightarrow E + T \cdot$

E

\$

Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \mathbf{int}$

$T \rightarrow (E)$

$S \rightarrow \cdot E\$$

E

\$

Tracking our Position in Productions

$S \rightarrow E\$$

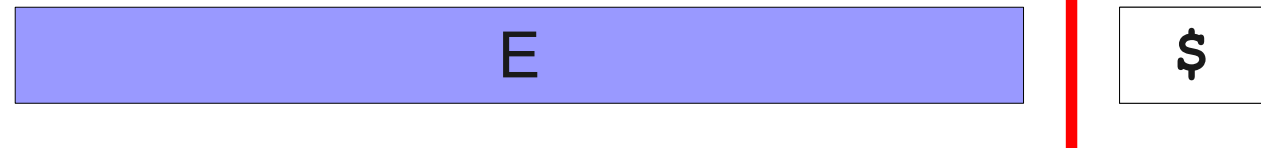
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \mathbf{int}$

$T \rightarrow (E)$

$S \rightarrow E \cdot \$$



Tracking our Position in Productions

$S \rightarrow E\$$

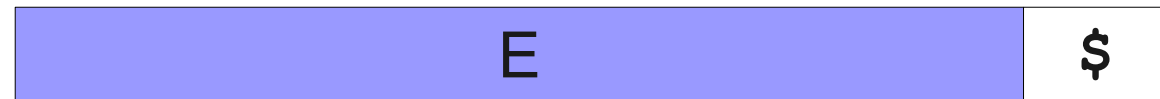
$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \mathbf{int}$

$T \rightarrow (E)$

$S \rightarrow E \cdot \$$



Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \text{int}$

$T \rightarrow (E)$

$S \rightarrow E \$ \cdot$

E \$



Tracking our Position in Productions

$S \rightarrow E\$$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow \mathbf{int}$

$T \rightarrow (E)$

$S \rightarrow E \$ \cdot$

S