

# First and Follow sets for TruPL v1.2

$\alpha$	First( $\alpha$ )	Follow( $\alpha$ )
<i>program</i>	<i>program</i>	
<b>identifier</b>	<b>identifier</b>	
<b>;</b>	<b>;</b>	
<b>:</b>	<b>:</b>	
<b>,</b>	<b>,</b>	
<i>int</i>	<i>int</i>	
<i>bool</i>	<i>bool</i>	
<i>begin</i>	<i>begin</i>	
<i>end</i>	<i>end</i>	
<i>procedure</i>	<i>procedure</i>	
<b>(</b>	<b>(</b>	
<b>)</b>	<b>)</b>	
<b>:=</b>	<b>:=</b>	
<i>if</i>	<i>if</i>	
<i>then</i>	<i>then</i>	
<i>else</i>	<i>else</i>	
<i>while</i>	<i>while</i>	
<i>loop</i>	<i>loop</i>	
<i>print</i>	<i>print</i>	
<i>relop</i>	<i>relop</i>	
<b>addop</b>	<b>addop</b>	
<b>mulop</b>	<b>mulop</b>	
<b>num</b>	<b>num</b>	
<i>not</i>	<i>not</i>	
<b>PROGRAM</b>	<i>program</i>	<eof>
<b>DECL_LIST</b>	$\lambda, \text{procedure}, \text{identifier}$	<i>begin</i>
<b>VARIABLE_DECL_LIST</b>	$\lambda, \text{identifier}$	<i>procedure, begin</i>
<b>VARIABLE_DECL</b>	<b>identifier</b>	<b>;</b>

PROCEDURE_DECL_LIST	$\lambda, \textit{procedure}$	$\textit{begin}$
IDENTIFIER_LIST	$\textit{identifier}$	$:$
IDENTIFIER_LIST_PRIME	$\lambda, ,$	$:$
STANDARD_TYPE	$\textit{int}, \textit{bool}$	$;, )$
BLOCK	$\textit{begin}$	$;, \textit{else}$
PROCEDURE_DECL	$\textit{procedure}$	$;$
PROCEDURE_ARGS	$\lambda, \textit{identifier}$	$)$
FORMAL_PARM_LIST	$\textit{identifier}$	$)$
FORMAL_PARM_LIST_HAT	$\lambda, ;$	$)$
STMT_LIST	$;, \textit{identifier}, \textit{if}, \textit{while}, \textit{print}$	$\textit{end}$
STMT_LIST_PRIME	$\lambda, \textit{identifier}, \textit{if}, \textit{while}, \textit{print}$	$\textit{end}$
STMT	$\textit{identifier}, \textit{if}, \textit{while}, \textit{print}$	$;$
ASSIGNMENT_STMT	$\textit{identifier}$	$;$
IF_STMT	$\textit{if}$	$;$
IF_STMT_HAT	$\lambda, \textit{else}$	$;$
WHILE_STMT	$\textit{while}$	$;$
PRINT_STMT	$\textit{print}$	$;$
PROCEDURE_CALL_STMT	$\textit{identifier}$	$;$
EXPR_LIST	$\lambda, \textit{identifier}, \textit{num}, (, \textit{not}, +, -$	$)$
ACTUAL_PARM_LIST	$\textit{identifier}, \textit{num}, (, +, -, \textit{not}$	$)$
ACTUAL_PARM_LIST_HAT	$\lambda, ,$	$)$
EXPR	$\textit{identifier}, \textit{num}, (, +, -, \textit{not}$	$;, \textit{then}, \textit{loop}, ,, )$
EXPR_HAT	$\lambda, \textit{relop}$	$;, \textit{then}, \textit{loop}, ,, )$
SIMPLE_EXPR	$\textit{identifier}, \textit{num}, (, +, -, \textit{not}$	$\textit{relop}, ;, \textit{then}, \textit{loop}, ,, )$
SIMPLE_EXPR_PRIME	$\lambda, \textit{addop}$	$\textit{relop}, ;, \textit{then}, \textit{loop}, ,, )$
TERM	$\textit{identifier}, \textit{num}, (, +, -, \textit{not}$	$\textit{addop}, \textit{relop}, ;, \textit{then}, \textit{loop}, ,, )$
TERM_PRIME	$\lambda, \textit{mulop}$	$\textit{addop}, \textit{relop}, ;, \textit{then}, \textit{loop}, ,, )$
FACTOR	$\textit{identifier}, \textit{num}, (, +, -, \textit{not}$	$\textit{mulop}, \textit{addop}, \textit{relop}, ;, \textit{then}, \textit{loop}, ,, )$
SIGN	$+, -, \textit{not}$	$\textit{identifier}, \textit{num}, (, +, -, \textit{not}$