Γραμματική της EEL

\<program\>	::= **program** id \<block\> **endprogram**
\<block\>	::= \<declarations\> \<subprograms\> \<statements\>
\<declarations\>	::= ε | **declare** \<varlist\> **enddeclare**
\<varlist\>	::= ε | id (, id)*
\<subprograms\>	::= (\<procorfunc\>) *
\<procorfunc\>	::= **procedure** id \<procorfuncbody\> **endprocedure** |
	function id \<procorfuncbody\> **endfunction**
\<procorfuncbody\>	::= \<formalpars\> \<block\>
\<formalpars\>	::= **(** \<formalparlist\> **)**
\<formalparlist\>	::= \<formalparitem\> (, \<formalparitem\>)* | ε
\<formalparitem\>	::= **in** id | **inout** id
\<statements\>	::= \<statement\> (; \<statement\>)*
\<statement\>	::= ε |

<dl>
<dd>

                    \<assignment-stat\> |

                    \<if-stat\> |

                    \<while-stat\> |

                    \<repeat-stat\> |

                    \<exit-stat\> |

                    \<switch-stat\> |

                    \<forcase-stat\> |

                    \<call-stat\> |

                    \<return-stat\> |

                    \<input-stat\> |

                    \<print-stat\>

</dd>
</dl>

\<assignment-stat\>	::= id **:=** \<expression\>
\<if-stat\>	::= **if** \<condition\> **then** \<statements\> \<elsepart\> **endif**
\<elsepart\>	::= ε | **else** \<statements\>
\<repeat-stat\>	::= **repeat** \<statements\> **endrepeat**

Πανεπιστήμιο Ιωαννίνων
Πολυτεχνική Σχολή
Τμήμα Μηχανικών Η/Υ και Πληροφορικής
"Μεταφραστές"
Διδάσκων: Γ. Μανής
Φεβρουάριος 2018

<exit-stat>	::= **exit**
<while-stat>	::= **while** <condition> <statements> **endwhile**
<switch-stat>	::= **switch** <expression>
	(**case** <expression> **:** <statements>)+
	endswitch
<forcase-stat>	::= **forcase**
	(**when** <condition> **:** <statements>)+
	endforcase
<call-stat>	::= **call** id <actualpars>
<return-stat>	::= **return** <expression>
<print-stat>	::= **print** <expression>
<input-stat>	::= **input** id
<actualpars>	::= **(** <actualparlist> **)**
<actualparlist>	::= <actualparitem> (**,** <actualparitem>)* | ε
<actualparitem>	::= **in** <expression> | **inout** id
<return-stat>	::= **return** <expression>
<condition>	::= <boolterm> (**or** <boolterm>)*
<boolterm>	::= <boolfactor> (**and** <boolfactor>)*
<boolfactor>	::=**not [**<condition>**]** | **[**<condition>**]** |
	<expression> <relational-oper> <expression> |
	true | **false**
<expression>	::= <optional-sign> <term> (<add-oper> <term>)*
<term>	::= <factor> (<mul-oper> <factor>)*
<factor>	::= constant | **(**<expression>**)** | id <idtail>
<idtail>	::= ε | <actualpars>
<relational-oper>	::= **=** | **<=** | **>=** | **>** | **<** | **<>**
<add-oper>	::= **+** | **-**
<mul-oper>	::= ***** | **/**
<optional-sign>	::= ε | <add-oper>