(Enhanced) Grammar for Basic SPLAT 1.2

<program> ::= **program** <decls> **begin** <stmts> **end** **;**

<decls> ::= **(** <decl> **)***

<decl> ::= <var-decl>

| <func-decl>

<var-decl> ::= <label> **:** <type> **;**

<func-decl> ::= <label> **(** <params> **) :** <ret-type> **is** <loc-var-decls> **begin** <stmts> **end ;**

<params> ::= <param> **(** **,** <param> **)***

| ε

<param> ::= <label> : <type>

<loc-var-decls> ::= **(** <var-decl> **)***

<stmts> ::= **(** <stmt> **)***

<stmt> ::= <label> **:=** <expr> **;**

| **while** <expr> **do** <stmts> **end while** **;**

| **if** <expr> **then** <stmts> **else** <stmts> **end if** **;**

| **if** <expr> **then** <stmts> **end if** **;**

| <label> **(** <args> **)** **;**

| **print** <expr> **;**

| **print_line** **;**

| **return** <expr> **;**

| **return** **;**

<expr> ::= **(** <expr> <bin-op> <expr> **)**

| **(** <unary-op> <expr> **)**

| <label> **(** <args> **)**

| <label>

| <literal>

<bin-op> ::= **and** | **or** | **>** | **<** | **==** | **>=** | **<=** | **+** | **-** | ***** | **/** | **%**

<unary-op> ::= **not** | **-**

<args> ::= <expr> **(** **,** <expr> **)***

| ε

<label> ::= *...sequence of alphanumeric characters and underscore, not starting with a digit...*

<ret-type> ::= <type> | **void**

<type> ::= **Integer** | **Boolean** | **String**

<literal> ::= <int-literal> | <bool-literal> | <string-literal>

<int-literal> ::= *...sequence of decimal digits...*

<bool-literal> ::= **true** | **false**

<string-literal> ::= **"** *...sequence of non-double-quote, non-backslash displayable characters and spaces...* **"**