LFPC

Laboratory work **Nr.3**

V_T = {function, identifier, alpha, digit, type, body, parameters, declaration, expression, conditional statement, function invocation, assignment, literal, characters, etc.}

V_N = {**void**, **int**, **decimal**, **boolean**, **string**, **if**, **else**, **and**, **or**, **true**, **false**, **return**, **Unicode set**, **a** … **z**, **A** … **Z**, **0** … **1**}

P = {<function > → <type> <identifier> **(**<parameters>?**)** <body>

<identifier> → <alpha> $^+$ <digit>*<alpha>*

<alpha> → **a** | … | **z** | **A** | … | **Z**

<digit> → **0** | **1**| **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**

<non-zero digit> → **1**| **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**

<parameters> → <identifier>**,** <parameters> | <identifier>

<body> → **{**<declarations>?**}**

<declarations> → <declaration><declarations> | <declaration>

<declaration> → <type> <identifier> <assignment>**;** | <function invocation>**;** | **return** <expression> | <conditional statement>

<expression> → <conditional expression> | <simple expression>

<conditional statement> → <if statement>

<if statement> → **if** (<conditional expression>) <body> **else**? <body>?

<conditional expression> → <conditional and> $^+$ | <conditional or> $^+$ | <comparator expression>

<conditional and> → <comparator expression> **and** <comparator expression>

<conditional or> → <comparator expression> **or** <comparator expression>

<comparator expression> → <identifier> <relational operator> <identifier>

<relational operator> → **==** | **!=** | **>** | **>=** | **<** | **<=**

<function invocation> → <identifier> **(**<parameters>**)**

<assignment> → **=** < simple expression>? | **=** <function invocation>

<simple expression> → <literal> <operator> <expression> | <literal>

<operator> → **+** | **-** | ***** | **/**

<literal> → <integer literal> | <floating-point literal> | <string literal> | <boolean literal>

<integer literal> → <non-zero digit> <digit> $^+$

<decimal literal> → <non-zero digit> <digit> $^+$ **.** <digit> $^+$

<boolean literal> → **true** | **false**

<string literal> → "< characters>?"

<characters> → **Unicode set**

<type> → **void** | **int** | **decimal** | **boolean** | **string**

}