Laborator 1 – LTFC

1. **Textele sursa ale programelor:**

|  |  |  |
| --- | --- | --- |
| 1. Perimetrul si raza cercului de raza data | 1. CMMDC a 2 numere naturale date | 1. Suma a n numere reale citite de la tastatura |
| BEGIN real a , p , r , pi ; START pi = 3 . 14 ; read ( r ) ; a = pi \* r \* r ; p = 2 \* pi \* r ; write ( a ) ; write ( p ) ; STOP END | BEGIN int a , b ; START while ( a != b ) START if ( a > b ) START a = a - b ; STOP else START b = b - a ; STOP STOP write ( a ) ; STOP END | BEGIN int n , suma = 0 , numar ; START for ( int i = 1 ; i <= n ; i = i + 1 ) START read ( numar ) ; suma = suma + numar ; STOP write ( suma ) ; STOP END |

1. **Specificarea mini-limbajului:**
2. **Alfabetul limbajului:**
3. Litere mari ale alfabetului englez (A-Z)
4. Litere mici ale alfabetului englez (a-z)
5. Cifrele sistemului zecimal (0-9)
6. **Lexic:**
7. Simboluri pentru reprezentarea:

* Operatorilor: +, -, /, \*, %, <, >, <=, >=, !=, ==, =
* Separatorilor: ;, (, ), {, }, blank, :, ,, [, ]
* Cuvinte rezervate: read, write, if, else, begin, end, for, while, real, int, char, array, OR, AND, start, stop

1. Identificatori.Reguli lexicale:

* Identificator: litera | litera {litera} {cifra}
* Cifra:”0” | ”1” | ”2” | ”3” | ”4” | ”5” | ”6” | ”7” | ”8” | ”9”
* Relatie: „<” | ”<=” | ”!=” | ”>” | ”>=” | ”==”
* Integer: „0” [„+”|”-„] cifra\* {cifra}
* Cifra\*: ”1”|”2”|”3”|”4”|”5”|”6”|”7”|”8”|”9”
* Real: [„+”|”-„] {cifra} ”.” {cifra} cifra\* | {cifra}
* Const: real | integer | string

1. **Sintaxa mini-limbajului:**

<PROGRAM> 🡪 <INTRARE\_PROGRAM>

<LISTA\_DECLARATII>

<INTRARE\_BLOC>

<LISTA\_INSTRUCTIUNI>

<IESIRE\_BLOC>

<IESIRE\_PROGRAM>

* <INTRARE\_PROGRAM> 🡪 BEGIN
* <IESIRE\_PROGRAM> 🡪 END
* <INTRARE\_BLOC> 🡪START
* <IESIRE\_BLOC> 🡪 STOP
* <LISTA\_DECLARATII> 🡪 <DECLARATIE>;<LISTA\_DECLARATII> | <DECLARATIE>;
  + <DECLARATIE> 🡪 <TIP><LISTA\_VARIABILE> | <TIP><LISTA\_INSTRUCTIUNI\_ATRIBUIRE>
    - <TIP> 🡪 <TIP1> | <TIP2>
      * <TIP1> 🡪 INT | FLOAT | CHAR
      * <TIP2> 🡪 ARRAY[<CONSTANTA>] <TIP1>
    - <LISTA\_VARIABILE> 🡪 <VARIABILA>,<LISTA\_VARIABILE> | <VARIABILA>
    - <LISTA\_INSTRUCTIUNI\_ATRIBUIRE> 🡪 <INSTRUCTIUNE\_ATRIBUIRE> | <INSTRUCTIUNE\_ATRIBUIRE>,<LISTA\_INSTRUCTIUNI\_ATRIBUIRE>
* <LISTA\_INSTRUCTIUNI> 🡪 <INSTRUCTIUNE>;<LISTA\_INSTRUCTIUNI> | <INSTRUCTIUNE>;
  + <INSTRUCTIUNE> 🡪 <INSTRUCTIUNE\_ATRIBUIRE> | <INSTRUCTIUNE\_SELECTIE> | <INSTRUCTIUNE\_INPUT/OUTPUT> | <INSTRUCTIUNE\_CICLARE> | <INSTRUCTIUNE\_FOR>
    - <INSTRUCTIUNE\_ATRIBUIRE> 🡪 <VARIABILA> ‘=’ <EXPRESIE>
      * <EXPRESIE> 🡪 <TERMEN> <OPERATOR> ‘(‘ <EXPRESIE> ‘)’ | <TERMEN>
        + <TERMEN> 🡪 <CONSTANTA> | <VARIABILA>
        + <OPERATOR> 🡪 ‘+’ | ‘-‘ | ‘\*’ | ‘/’
    - <INSTRUCTIUNE\_CICLARE> 🡪 WHILE ‘(‘ <LISTA\_CONDITII> ‘)’ <INTRARE\_BLOC> <LISTA\_INSTRUCTIUNI> <IESIRE\_BLOC>
      * <LISTA\_CONDITII> 🡪 <CONDITIE> <OPERATOR\_LOGIC> <LISTA\_CONDITII> | <CONDITIE>
      * <CONDITIE> 🡪 <EXPRESIE><OPERATOR\_RELATIONAL><EXPRESIE>;
      * <OPERATOR\_RELATIONAL> 🡪 ‘<’ | ‘>’ | ‘=’ | ‘!=’ | ‘<=’ | ‘>=’
      * <OPERATOR\_LOGIC> 🡪 ‘AND’| ‘OR’
    - <INSTRUCTIUNE\_SELECTIE) 🡪 ‘IF(’ <LISTA\_CONDITII> ‘)’<INTRARE\_BLOC> <LISTA\_INSTRUCTIUNI> <IESIRE\_BLOC> ‘ELSE’ <INTRARE\_BLOC> <LISTA\_INSTRUCTIUNI> <IESIRE\_BLOC> | ‘IF(’ <LISTA\_CONDITI> ‘)’<INTRARE\_BLOC> <LISTA\_INSTRUCTIUNI><IESIRE\_BLOC>
    - <INSTRUCTIUNE\_INPUT/OUTPUT> 🡪 ‘read(’ <VARIABILA> ‘);’ | ‘write(’ <VARIABILA> | <CONSTANTA> ‘);’
    - <INSTRUCTIUNE\_FOR> 🡪 ‘for(’<LISTA\_DECLARATII> <LISTA\_CONDITII> <LISTA\_INSTRUCTIUNI\_ATRIBUIRE> ‘)’<INTARE\_BLOC> <LISTA\_INSTRUCTIUNI> <IESIRE\_BLOC>

1. **Codificarea:**

|  |  |
| --- | --- |
| **Tip atom** | **Cod** |
| constanta | 0 |
| variabila | 1 |
| begin | 2 |
| end | 3 |
| start | 4 |
| stop | 5 |
| if | 6 |
| while | 7 |
| for | 8 |
| write | 9 |
| read | 10 |
| else | 11 |
| int | 12 |
| real | 13 |
| char | 14 |
| ; | 15 |
| ( | 16 |
| ) | 17 |
| , | 18 |
| + | 19 |
| - | 20 |
| \* | 21 |
| / | 22 |
| % | 23 |
| > | 24 |
| < | 25 |
| >= | 26 |
| <= | 27 |
| = | 28 |
| == | 29 |
| AND | 30 |
| OR | 31 |
| . | 32 |
| != | 33 |