Gramatyka języka

```
A \in Alpha ::= a|b|c|d|e|f|g|h|i|j|k|||m|n|o|p|r|s|t|u|w|v|x|y|z|_
st \in String ::= Ast | \epsilon
d \in Digit ::= 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
N \in Num := d0N \mid dN \mid \epsilon
sg ∈ Sign ::= - | ε
i \in Int ::= sgN
x \in Var ::= st
p ∈ IDE ::= st
e \in Exp := i | x | e_1 + e_2 | e_1 * e_2 | e_1 - e_2 | e_1/e_2 | (e) | not e | e_1 and e_2 | e_1 or e_2 | e_1 = < e_2 |
e_1 \le e_2 | e_1 \le e_2 | e_1 \ge e_2 | e_1 = e_2
S \in \mathbf{Stmt} ::= x := e \mid \mathbf{skip} \mid S_1; S_2 \mid \text{if e then } S_1 \text{ else } S_2 \text{ fi} \mid \text{if e then } S \text{ fi} \mid \text{while e do } S \mid \text{begin}
S end | call p(D_a) | for x := e_1 to e_2 do S | for x := e_1 downto e_2 do S | print x
D_v \in VDecl ::= var x; D_v \mid \epsilon
D_a \in ADecl ::= vr x; D_a | vl x; D_a | \epsilon
D_p \in PDecl ::= procedure p(D_a) S; D_p \mid \varepsilon
Prog ::= program D<sub>v</sub> D<sub>p</sub> S
                                             Przykładowy program
słowa kluczowe zaznaczone są na czerwono, przykładowy program:
program
var wm;
var wd;
var x;
var y;
procedure mnoz(vr w; vl x; vl y)
        w := x^*y;
procedure dziel(vr w; vl x; vl y)
        w := x/y;
begin
        for i := 1 \text{ to } 10 \text{ do}
        begin
                 x := 2*i;
                 y := i;
```

call mnoz(wm, x,y);
call dziel(wd,x,y);

Program sprawdza czy procedury wykonują się tak jak powinny, przekazujemy wynik przez referencję, a więc odpowiednio na wm i wd powinien zostać przekazany wynik.

Opis języka

Język zawiera konstrukcje typowe dla pascala, w przypadku:

- procedur
- pętli while
- petli for
- instrukcji if, literał fi mówi gdzie kończą się warunki podane przez if
- bloków instrukcji → begin ... end
- operatorów przypisań, operatorów logicznych: "=<", "=", ":=", "not", "or", "and"

Ponadto język zawiera deklarację wywołania procedur "call nazwa_procedury(argumenty)" oraz deklarację argumentów, która zawiera deklaracje zmiennych przekazywanych przez wartość – "vl x" oraz deklaracje zmiennych przekazywanych przez referencję – "vr x". Dodatkowo występuje instrukcja "print x", która wypisuje na standardowe wyjście wartość zmiennej x. Jedynym typem danych w języku jest typ liczb całkowitych – Int. Wyrażenia w if, while są prawdziwe, gdy mają wartość różną od zera, fałszywe gdy mają wartość równą zero. Wyrażenia logiczne ($e_1 = e_2$, $e_1 = < e_2$) zwracają wartość różną od zera gdy są prawdziwe i wartość równą zero gdy są fałszywe. Operatory logiczne również działają standardowo (w logice dwuwartościowej dla liczb: różna od zera, równa zero).