

Реализация интерпретатора модельного языка

Задача: разработать и реализовать интерпретатор модельного языка программирования. Инструментальный язык - C++.

Этапы:

- лексический анализ
- синтаксический анализ
- семантический анализ
- генерация программы на внутреннем языке (ПОЛИЗ)
- интерпретация программы на внутреннем языке

Описание модельного языка:

$P \rightarrow \text{program } D1; B@$

$D1 \rightarrow \text{var } D\{,D\}$

$D \rightarrow I\{,I\}:[\text{int} \mid \text{bool} \mid \text{real}]$

$B \rightarrow \text{begin } S \{;S\} \text{ end}$

$S \rightarrow I:=E \mid \text{if } E \text{ then } S \text{ else } S \mid \text{while } E \text{ do } S \mid B \mid \text{read}(I) \mid \text{write}(E) \mid \text{if } E \text{ then } S \mid$
 $\text{repeat } S \text{ until } E \mid \text{break}$

$E \rightarrow E1[= \mid < \mid > \mid \leq \mid \geq \mid !=] E1 \mid E1$

$E1 \rightarrow T\{ [+ \mid - \mid \text{or}] T\}$

$T \rightarrow F\{ [* \mid / \mid \text{and}] F\}$

$F \rightarrow I \mid N \mid L \mid V \mid \text{not } F \mid (E) \mid + F \mid - F$

$L \rightarrow \text{true} \mid \text{false}$

$I \rightarrow C \mid IC \mid IR$

$C \rightarrow a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z|$
 $|A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z$

$N \rightarrow R \{R\}$

$V \rightarrow X.Y$

$X \rightarrow R \{R\}$

$Y \rightarrow R \{R\}$

$R \rightarrow 0|1|2|3|4|5|6|7|8|9$

Описание с помощью БНФ:

<программа> → **program** <описания>; <операторные скобки>@
<описания> → **var** <описание> { ,<описание>}
<описание> → <переменная> { ,<переменная>} : <тип>
<тип> → **int** | **bool** | **real**
<операторные скобки> → **begin** <оператор> { ; <оператор>} **end**
<оператор> → <переменная> := <выражение>
| **if** <выражение> **then** <оператор> **else** <оператор>
| **while** <выражение> **do** <оператор>
| <операторные скобки>
| **read** (<переменная>) | **write**(<выражение>)
| **if** <выражение> **then** <оператор>
| **repeat** <оператор> **until** <выражение>
| **break**
<выражение> → <выражение1>[= | < | > | <= | >= | !=] <выражение1> | <выражение1>
<выражение1> → <выражение2>{ [+ | - | **or**] <выражение2>}
<выражение2> → <константа> { [* | / | **and**] <константа>}
<константа> → <строковая> | <целочисленная> | <логическая> | <вещественная>
| **not** <константа> | (<выражение>) | + <константа> | - <константа>
<логическая> → **true** | **false**
<строковая> → <литера> | <строковая> <литера> | <строковая> <цифра>
<литера> → a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z|
| A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z|
<целочисленная> → <цифра> { <цифра>}
<вещественная> → <целая часть>.<дробная часть>
<целая часть> → <цифра> { <цифра>}
<дробная часть> → <цифра> { <цифра>}
<цифра> → 0|1|2|3|4|5|6|7|8|9

Семантика:

Любой идентификатор, используемый в программе, должен быть описан и только один раз.

Семантика операторов *if-then-else* , *while-do* , *repeat-until* как в Паскале, в выражении-условии допустим только логический тип; оператор *read* — оператор ввода значения переменной; *write* — оператор вывода значения выражения, указанного в круглых скобках. *Break* - оператор выхода из цикла.

Числовые константы записываются в десятичной системе счисления.

Набор операций и их старшинство:

not + — (логическое отрицание, унарные плюс и минус)

**/ and* (умножение, деление, логическое умножение)

+ — *or* (сложение, вычитание, логическое сложение)

< > <= >= != (операции отношения)

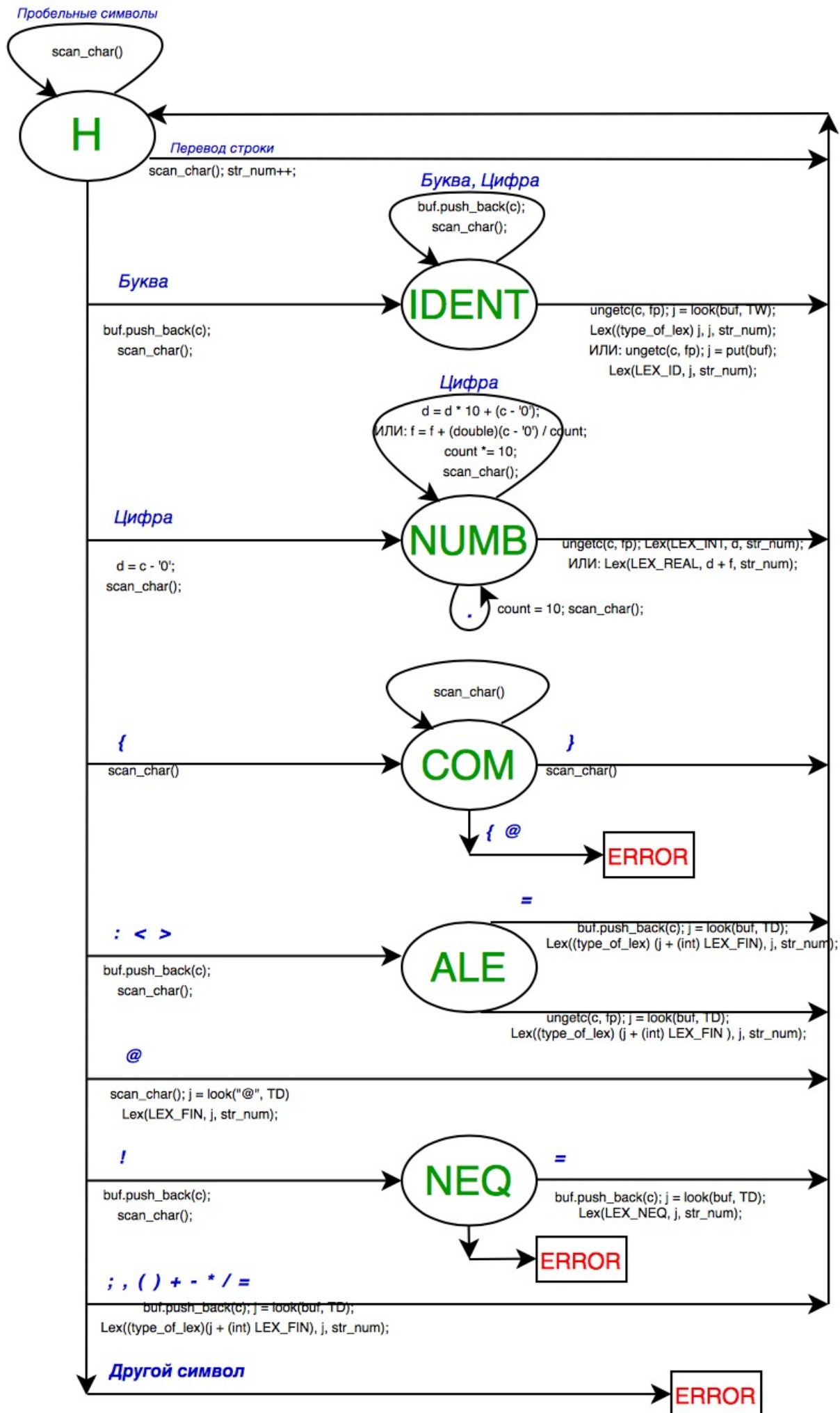
:= (присваивание)

Семантика операций общепринятая (семантика операции присваивания как в Паскале). Вычисления проводятся по обычной схеме: слева направо, вычисление всего выражения (как в Паскале).

Таблица типов в результате выполнения операций:

Операция	Тип X	Тип Y	Тип результата
+ - * /	int	int	int
+ - * /	int	real	real
+ - * /	real	int	real
+ - * /	real	real	real
унарный +	int	-	int
унарный -	real	-	real
< > <= >= !=	int	int	bool
< > <= >= !=	int	real	bool
< > <= >= !=	real	int	bool
< > <= >= !=	real	real	bool
and or	bool	bool	bool
not	bool	-	bool
:=	int	int	int
:=	int	real	int
:=	real	int	real
:=	real	real	real
:=	bool	bool	bool

Конечный автомат с действиями, лежащий в основе лексического анализатора модельного языка:



Исходный код программы Интерпретатора:

https://github.com/yanaredkina/Interpreter_Project

Комплекты тестов:

https://github.com/yanaredkina/Interpreter_Project

В А-тестах (корректные программы) проверяются:

test1 - деление и сравнение чисел, цикл REPEAT, оператор вывода WRITE;

test2 - операторы ввода/вывода, условный оператор, унарный минус;

test3 - вычисление чисел Фибоначчи;

test4 - цикл WHILE, оператор BREAK;

test5 - приведение типов в операторе присваивания и арифметических операциях.

В В-тестах (программы с ошибками) проверяются:

test1 - лексическая ошибка;

test2 - синтаксическая ошибка;

test3 - семантическая ошибка;

test4 - семантическая ошибка;

test5 - деление на ноль.