

Факультет ИУ "Информатика и системы управления"

Кафедра ИУ-3 "Информационные системы и телекоммуникации"

Отчет по лабораторной работе №4

по курсу "Представление знаний в информационных системах"

направления 09.04.02 (магистр)

" Реализация синтаксического разбора на языке программирования Пролог"

Продолжительность 4 часа.

Выполнил:

студент группы ИУ3-22М

Быковский М. К.

Проверил:

Лычков И. И.

Цель работы

Цель работы – изучить возможности языка Пролог для решения задач синтаксического разбора текстов.

Текст задания по варианту

Разработать программу на Прологе для разбора арифметических выражений. Программа должна учитывать приоритеты арифметических операций.

Пример 1. Исходные данные для разбора: $a*2+b*(c-1)$

Представление исходных данных в Прологе:

word(a, 1, 2).

word(times, 2, 3).

word(const, 3, 4).

word(plus, 4, 5).

word(b, 5, 6).

word(times, 6, 7).

word(leftbracket, 7, 8).

word(c, 8, 9).

word(minus, 9, 10).

word(const, 10, 11).

word(rightbracket, 11, 12).

Результат разбора:

sum(prod(a, const), prod(b, dif(c, const)))

Пример 2. Исходные данные для разбора: $a*2+b$

Представление исходных данных в Прологе:

word(a, 1, 2).

word(times, 2, 3).

word(const, 3, 4).

word(plus, 4, 5).

word(b, 5, 6).

Результат разбора:

```
sum(prod(a, const), b)
```

Пример 3. Исходные данные для разбора: $a+2*b$

Представление исходных данных в Прологе:

```
word(a, 1, 2).
```

```
word(plus, 2, 3).
```

```
word(const, 3, 4).
```

```
word(times, 4, 5).
```

```
word(b, 5, 6).
```

Результат разбора:

```
sum(a, prod(const, b))
```

Примечание. Для разбора вложенных бинарных операций используйте правила вида:

```
binop(P1,P4,Result) :- operand(P1,P2,Op1), word(plus, P2, P3), operand(P3, P4, Op2), format(Result, ...).
```

```
operand(P1, P2, Result) :- binop(P1, P2, Result). % правило для случая  
вложенной операции
```

```
operand(P1, P2, Result) :- ... . % правило для простого операнда
```

Для защиты от переполнения стека необходимо ограничить глубину поиска (см. замечания к варианту 1).

Листинг разработанной программы на Прологе

```
%word(a, 1, 2).  
%word(times, 2, 3).  
%word(const, 3, 4).  
%word(plus, 4, 5).  
%word(b, 5, 6).  
%  
%  
%word(a, 1, 2).  
%word(plus, 2, 3).  
%word(const, 3, 4).
```

```

%word(times, 4, 5).
%word(b, 5, 6).
%
%
word(a, 1, 2).
word(times, 2, 3).
word(const, 3, 4).
word(plus, 4, 5).
word(b, 5, 6).
word(times, 6, 7).
word(leftbracket, 7, 8).
word(c, 8, 9).
word(minus, 9, 10).
word(const, 10, 11).
word(rightbracket, 11, 12).

strcat6(Str1, Str2, Str3, Str4, Str5, Str6, Result):-
    string_concat(Str1, Str2, Str12),
    string_concat(Str12, Str3, Str123),
    string_concat(Str123, Str4, Str1234),
    string_concat(Str1234, Str5, Str12345),
    string_concat(Str12345, Str6, Result).

strcat4(Str1, Str2, Str3, Result):-
    string_concat(Str1, Str2, Str12),
    string_concat(Str12, Str3, Result).

% Грамматические правила
%
binop(P1,P4,Result,N) :- leftbrac(P1, P2, Op1), operand(P2, P3, N, Op2),
    rightbrac(P3,P4,Op3), strcat4("leftbracket(", Op2, ")rightbracket", Result).
binop(P1,P4,Result,N) :- operand(P1, P2, N, Op1), word(Word, P2, P3),
    Word==plus, operand(P3,P4,N,Op2), strcat6(Word, "(", Op1, ", ", Op2, ")",
    Result).
binop(P1,P4,Result,N) :- operand(P1, P2, N, Op1), word(Word, P2, P3),
    Word==minus, operand(P3,P4,N,Op2), strcat6(Word, "(", Op1, ", ", Op2, ")",
    Result).
binop(P1,P4,Result,N) :- operand(P1, P2, N, Op1), word(Word, P2, P3),
    Word==times, operand(P3,P4,N,Op2), strcat6(Word, "(", Op1, ", ", Op2, ")",
    Result).

```

```

operand(P1, P2, CurrLevel, Result) :- CurrLevel > 10, !, fail.
operand(P1, P2, CurrLevel, Result) :- const(P1, P2, Result).
operand(P1, P4, CurrLevel, Result) :- NextLevel = CurrLevel+1, binop(P1, P4,
Result,NextLevel).

% Объявления слов, отнесенных к частям речи

isconst(a).
isconst(b).
isconst(c).
isconst(const).

isleftbrac(leftbracket).
isrightbrac(rightbracket).

const(From, To, Word) :- word(Word, From, To), isconst(Word).
leftbrac(From, To, Word) :- word(Word, From, To), isleftbrac(Word).
rightbrac(From, To, Word) :- word(Word, From, To), isrightbrac(Word).

```

Скриншоты с результатами выполнения программы

$a*2+b*(c-1)$

binop(1, 12, Rez, 0).

Singleton variables: [Op1,Op3]

Singleton variables: [P1,P2,Result]

Singleton variables: [CurrLevel]

Rez = "plus(times(a, const), times(b, leftbracket(minus(c, const))rightbracket))"

7.257 seconds cpu time

Next 10 100 1,000 Stop

?- binop(1, 12, Rez, 0).

$a*2+b$

binop(1, 6, Rez, 0).

Singleton variables: [Op1,Op3]

Singleton variables: [P1,P2,Result]

Singleton variables: [CurrLevel]

Rez = "plus(times(a, const), b)"

1.156 seconds cpu time

Next 10 100 1,000 Stop

?- binop(1, 6, Rez, 0).

$a+2*b$

binop(1, 6, Rez, 0).

Singleton variables: [Op1,Op3]

Singleton variables: [P1,P2,Result]

Singleton variables: [CurrLevel]

Rez = "plus(a, times(const, b))"

0.503 seconds cpu time

Next 10 100 1,000 Stop

?- binop(1, 6, Rez, 0).