

#### Лабораторная работа №4.

### Изучение алгоритмов грамматического разбора предложений.

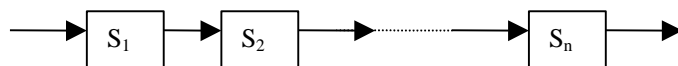
**Цель работы:** Изучение алгоритма грамматического разбора для заданного синтаксиса.

**Содержание работы:** В процессе выполнения лабораторной работы должны быть изучены структуры синтаксических графов и возможность их применения для задачи нисходящего распознавания предложений языка для заданного синтаксиса.

Заданный синтаксис некоторого языка удобно представлять в виде синтаксического графа, или графа распознавания. Такой граф отражает управление ходом работы при грамматическом анализе предложений. Синтаксический граф является эквивалентным представлением грамматики языка, его можно использовать вместо множества порождающих правил БНФ. Это очень удобная форма, и во многих случаях она оказывается предпочтительнее БНФ. Кроме того, граф дает более ясное и точное представление о структуре языка, а также позволяет лучше отобразить процесс грамматического разбора. Алгоритм, который распознает предложения некоторого языка, можно построить на основе его синтаксического графа (если такой граф существует). Такой граф фактически представляет собой или блок – схему алгоритма, или структуру данных, управляющую работой алгоритма. В первом случае говорят об алгоритме грамматического разбора для заданного синтаксиса, а во втором – о таблично – управляемом алгоритме. В настоящей лабораторной работе изучается алгоритм грамматического разбора для заданного синтаксиса. Преобразование графа в блок – схему алгоритма подчиняется определенным правилам. Эти правила таковы:

Пусть оператор, полученный с помощью преобразования графа  $S$ , обозначается через  $T(S)$ .

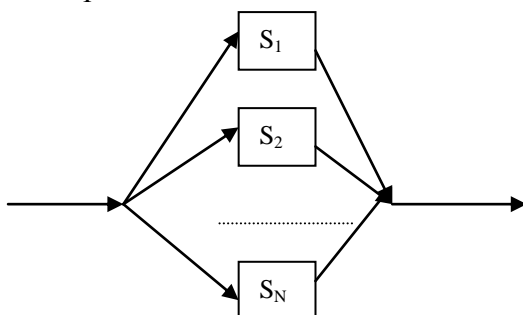
1. Необходимо свести систему графов к как можно меньшему числу отдельных графов с помощью соответствующих подстановок.
2. Необходимо преобразовать каждый граф в описание процедуры в соответствии с нижеприведенными правилами 3 – 7.
3. Последовательность элементов графа



преобразуется в составной оператор следующего вида:

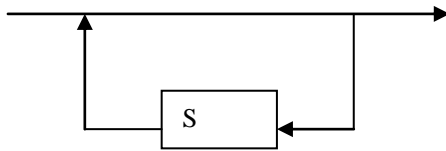
$\{ T(S_1); T(S_2); \dots T(S_N) \}$  Здесь  $\{$  и  $\}$  – операторные скобки. Могут быть begin – end.

4. Выбор элементов



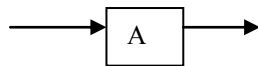
переводится в оператор выбора типа Case или switch или условный оператор if.

5. Цикл вида



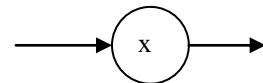
переводится в оператор цикла типа while.

6. Элемент графа, обозначающий другой граф A



переводится в оператор обращения к процедуре A.

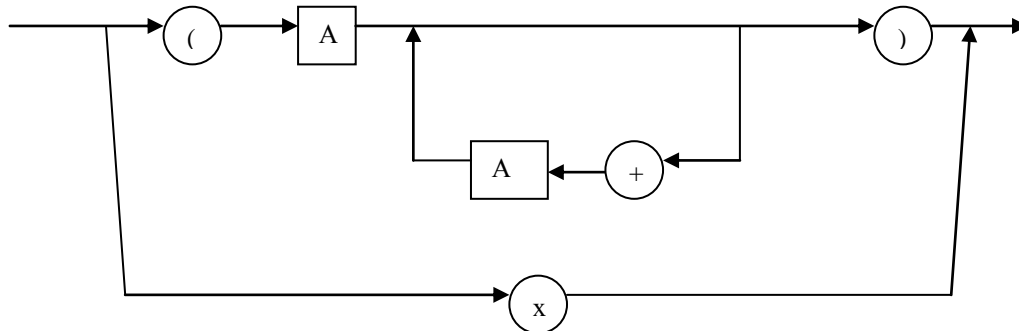
7. Элемент графа, обозначающий терминальный символ



переводится в оператор вида: if ch = x then read(ch) else error(); где ch – очередной прочитанный терминальный символ.

Применение правил 1 – 7 иллюстрируется следующим примером.

Пусть имеется синтаксический граф следующего вида:



Здесь «x», «(», «+», «)» - терминальные символы. Язык, соответствующий данному графу, состоит из предложений вида: x, (x), (x+x), ((x)) и т. п. Применение правил 1 – 7 к этому графу дает следующий алгоритм.

Parse

ch: char;

Procedure A;

begin

if ch = 'x' then read(ch)

```

else
if ch = '(' then
begin
read(ch);
A;
while ch = '+' do
begin
read(ch);
A
end;
if ch = ')' then read(ch) else error()
end else error()
end; //A

```

Основной блок алгоритма Parse:

```

read(ch);
A.

```

Некоторые комментарии к данному алгоритму:

1. Алгоритм имеет рекурсивную структуру;
2. Операция read(ch) означает чтение очередного символа из входной последовательности;
3. Считается, что терминальные символы имеют длину один байт;
4. Процедура error() предназначена для обработки синтаксических ошибок.

### **Порядок выполнения лабораторной работы.**

Для выполнения настоящей лабораторной работы необходимо:

1. В соответствии с полученным заданием в виде синтаксического графа некоторого языка построить алгоритм грамматического разбора согласно правилам 1 – 7;
2. По заданному синтаксическому графу построить несколько предложений языка, в том числе и неправильное, и проанализировать работу алгоритма.
3. Разработать приложение, реализующее построенный алгоритм. Приложение должно читать предложения, вводимые с клавиатуры и выполнять их грамматический разбор. В случае ошибки в исходном предложении должна выдаваться информация о ней, например, в виде местонахождения неправильной конструкции, а работа приложения – завершаться.

### **Содержание отчета.**

Отчет по лабораторной работе должен включать в себя:

1. Краткий листинг разработанного приложения с комментариями (В части реализации разработанных алгоритмов).
2. Заданный синтаксический граф и контрольные примеры в соответствии с п. 1 порядка выполнения лабораторной работы.
3. Пошаговую работу алгоритма для случаев правильного и неправильного исходных предложений в соответствии с контрольными примерами по п. 2.

### **Литература:**

1. Н. Вирт. «Алгоритмы + структуры данных = программы». М.: «Мир», 1985 г.