



ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
УПРАВЛЕНИЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ И ПОВЫШЕНИЯ
КВАЛИФИКАЦИИ

Кафедра «Программное обеспечение вычислительной техники и
автоматизированных систем»

Учебно-методическое пособие по дисциплине

«Теория языков программирования и методы трансляции»

Автор
Кудинов Н. В.

Ростов-на-Дону, 2019



Аннотация

Учебно-методическое пособие
предназначено для студентов очных форм
обучения направлений 09.04.04.

Авторы

К.т.н.,
доцент каф. ПОВТиАС
Кудинов Н.В.



Оглавление

1. Практическая работа №1	4
1.1. Цель работы	4
1.2. Задание	4
1.3. Ход работы	4
1.4. Содержание отчёта:	5
2. Практическая работа №2	5
2.1. Цель работы	5
2.2. Задание	5
2.3. Ход работы	5
2.4. Содержание отчёта:	6
3. Практическая работа №3	6
3.1. Цель работы	6
3.2. Задание	6
3.3. Ход работы	7
3.4. Содержание отчёта:	7
4. Практическая работа №4	7
4.1. Цель работы	7
4.2. Задание	8
4.3. Ход работы	8
4.4. Содержание отчёта:	8
5. Практическая работа №5	9
5.1. Цель работы	9
5.2. Задание	9
5.3. Ход работы	9
5.4. Содержание отчёта:	13
6. Практическая работа №6	13
6.1. Цель работы	13
6.2. Задание	14
6.3. Ход работы	14
6.4. Содержание отчёта:	16
7. Практическая работа №7	17
7.1. Цель работы	17
7.2. Задание	17
7.3. Ход работы	17
7.4. Содержание отчёта:	18

1. ПРАКТИЧЕСКАЯ РАБОТА №1

Грамматический вывод при изучении структуры искусственного языка

1.1. Цель работы

Получение представления о генеративной лингвистике. Получение навыков программирования формального вывода грамматически корректных цепочек языка. Оценка устойчивости алгоритма вывода.

1.2. Задание

Написать программу реализующую вывод терминальных цепочек из начального символа грамматики.

1.3. Ход работы

1. Выбрать из примеров грамматик, описанных в рекомендуемой литературе описание грамматики так, чтобы для неё был выписан вывод цепочек.

2. Осмыслить алгоритм вывода и реализацию процедур определения типов символов и выбора потенциальных правил.

3. Обдумать неоднозначность алгоритма и способы разрешения проблемы многозначности.

4. Реализовать алгоритм на одном из алгоритмических языков программирования.

5. Реализовать вывод корректных цепочек, сравнить с примерами из книг.

1.4. Содержание отчёта:

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.

2. ПРАКТИЧЕСКАЯ РАБОТА №2

Грамматический вывод при изучении структуры искусственного языка

2.1. Цель работы

Получение представления о генеративной лингвистике. Получение навыков программирования формального вывода грамматически корректных цепочек языка. Оценка устойчивости алгоритма вывода.

2.2. Задание

Написать программу реализующую вывод терминальных цепочек из начального символа грамматики.

2.3. Ход работы

1. Выбрать из примеров грамматик, описанных в рекомендуемой литературе описание грамматики так, чтобы для неё был выписан вывод цепочек.

2. Осмыслить алгоритм вывода и реализацию процедур

определения типов символов и выбора потенциальных правил.

3. Обдумать неоднозначность алгоритма и способы разрешения проблемы многозначности.

4. Реализовать алгоритм на одном из алгоритмических языков программирования.

5. Реализовать вывод корректных цепочек, сравнить с примерами из книг.

2.4. Содержание отчёта:

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.

3. ПРАКТИЧЕСКАЯ РАБОТА №3

Моделирование элементов современных языков программирования

3.1. Цель работы

Изучить способы формализации синтаксических конструкций языков программирования. Получить навыки лингвистического моделирования.

3.2. Задание

Составить грамматики для некоторых синтаксических кон-

струкций изучаемых языков.

3.3. Ход работы

1. Разобрать примеры описания списков, основных алгоритмических операторов, математических выражений.

2. Оценить сложность описания выбранных синтаксических конструкций.

3. Вспомнить особые синтаксические конструкции языков программирования изучаемых в вузе.

4. Выписать различные варианты написания этих элементов.

5. Разложить синтаксические элементы на более мелкие составляющие. Зафиксировать схему разложения.

6. Записать схему грамматики, построить синтаксическую диаграмму, итерационную форму.

3.4. Содержание отчёта:

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.

4. ПРАКТИЧЕСКАЯ РАБОТА №4

Моделирование машины Тьюринга при решении задачи обработки текстовой информации

4.1. Цель работы

Закрепить базовые знания по теории языков программирования, получить навыки моделирования абстрактного вычислителя на примере машины Тьюринга.

4.2. Задание

Составить грамматики для некоторых синтаксических конструкций изучаемых языков.

4.3. Ход работы

1. Найти в рекомендованной литературе формальное описание машины Тьюринга с примерами записей входной ленты и разбором очерёдности смены конфигураций.

2. Ввести параметры модели машины Тьюринга в программную модель программы JFLAP.

3. Реализовать смену конфигураций и записать последовательность конфигураций в отчёт, снабдив описание процесса номерами команд автомата.

4. Ознакомиться с пользовательским интерфейсом утилиты sed.

6. Реализовать процесс чтения и изменения ячеек машины Тьюринга на языке sed.

7. Проверить и отладить процесс работы машины.

8. Сделать выводы по практической работе.

4.4. Содержание отчёта:

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.

5. ПРАКТИЧЕСКАЯ РАБОТА №5

Синтаксический анализ. Автоматическое построение и реализация распознавателя-анализатора

5.1. Цель работы

Изучить современные методы создания транслирующих и интерпретирующих программ, получить навыки повторного использования универсального распознающего кода формируемого утилитами flex и bison.

5.2. Задание

Изучить результаты работы программного кода, полученного в результате обработки программами flex и bison формального описания грамматики языка.

5.3. Ход работы

1. Загрузить компьютер в Unix-совместимую ОС (в ОС Windows активизировать запуск эмулятора msys2, cygwin).
2. Сформировать в текущем каталоге файловой системы файлы, характеризующееся именем и содержанием, приведенным в табл 1.

Sample.txt

```
procedure Validate {  
  begin {  
    DoThis  
    Dothat  
  }  
  end {  
    CleanUp  
  }  
}
```

Parser.y

```
%{
#include <stdio.h>
%}

// Symbols.
%union
{
    char      *sval;
};
%token <sval> IDENTIFIER
%token PROCEDURE
%token BLOCK
%token ENDBLOCK

%start Procedure
%%

Procedure:
    PROCEDURE IDENTIFIER BLOCK { printf("Procedure : %s\n", $2); }
    Parts
    ENDBLOCK
    ;

Parts:
    /* empty */
    | Parts Part
    ;

Part:
    IDENTIFIER BLOCK { printf("\tPart : %s\n", $1); }
        Keywords
    ENDBLOCK
    ;

Keywords:
    /* empty */
    | Keywords Keyword
    ;

Keyword:
    IDENTIFIER { printf("\tKeyword : %s\n", $1); }
    ;
%%

int yyerror(char *s) {
    printf("yyerror : %s\n",s);
}

int main(void) {
    yyparse();
}
```

```

Parser.lex
%{
#include "Parser.h"
%}

blanks      [ \t\n]+
identifier  [_a-zA-Z0-9]+

%%

{blanks}    { /* ignore */ }

"procedure"          return(PROCEDURE);
"{"                  return(BLOCK);
"}"                  return(ENDBLOCK);

{identifier}        {
                    yyval.sval = malloc(strlen(yytext));
                    strncpy(yyval.sval, yytext,
strlen(yytext));
                    return(IDENTIFIER);
                }
    
```

3. Введите последовательность команд в терминале ОС, активизирующих последовательные процессы генерации синтаксического анализатора:

```

bison -d Parser.y
mv Parser.tab.h Parser.h
mv Parser.tab.c Parser.y.c
flex Parser.lex
mv lex.yy.c Parser.lex.c
gcc -g -c Parser.lex.c -o Parser.lex.o
gcc -g -c Parser.y.c -o Parser.y.o
gcc -g -o Parser Parser.lex.o Parser.y.o -lfl
    
```

При выполнении данных команд были созданы исходные файлы синтаксического анализатора (СА), объектные файлы и программа.

4. Проверить работоспособность СА, введя команду: Parser < sample.txt
5. Убедиться в том, что СА определил begin и end как Part, а

DoThis, DoThat и CleanUp как Keyword:

```
Procedure : Validate
  Part : begin
    Keyword : DoThis
    Keyword : Dothat
  Part : end
  Keyword : CleanUp
```

6. Сделать вывод о функциональном назначении flex и bison и о возможности внедрить CA в разрабатываемый программный модуль.

5.4. Содержание отчёта

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.

6. ПРАКТИЧЕСКАЯ РАБОТА №6

Семантический анализ и преобразование структур таблиц, представленных текстовыми файлами

6.1. Цель работы

Изучить современные методы создания транслирующих и интерпретирующих программ на уровне семантики языка, получить навыки обработки форматированных текстовых данных интерпретатором языка awk.

6.2. Задание

Изучить примеры программ на awk. Написать программу, переставляющую столбцы csv файла.

6.3. Ход работы

1. Загрузить компьютер в Unix-совместимую ОС (в ОС Windows активизировать запуск эмулятора msys2, cygwin).
2. Последовательно ввести в файл awk программы следующие инструкции:

```
awk '{print "Welcome to awk command tutorial "'}
```

```
awk -F: '{print $1}' /etc/passwd
```

```
echo "Hello Tom" | awk '{$2="Adam"; print $0}'
```

```
TESTFILE: {print $1 " home at " $6}
```

```
awk -F: -f testfile /etc/passwd
```

```
awk 'BEGIN {print "The File Contents:"}
```

```
{print $0}' myfile
```

```
awk 'BEGIN {print "The File Contents:"}
```

```
{print $0}
```

```
END {print "File footer"}' myfile
```

```
awk -f myscript /etc/passwd
```

```
myscript
BEGIN {
print "Users and thier corresponding home"
print " UserName \t HomePath"
print " _____ \t _____ "
FS=":"
}
{
print $1 " \t " $6
}
END {
print "The end"
}
```

```
awk 'BEGIN{FS=":"; OFS="-"} {print $1,$6,$7}'
/etc/passwd
```

```
testfile
1235.96521
927-8.3652
36257.8157
```

```
awk -f myscript /etc/passwd
```

addresses
Person Name
123 High Street
(222) 466-1234
Another person
487 High Street
(523) 643-8754

```
awk 'BEGIN{print ARGV,ARGV[1]}' myfile
```

```
awk '  
BEGIN{  
print ENVIRON["PATH"]  
}'
```

3. Почитать подробности про переменные окружения awk-программы
4. Написать программу, переставляющую столбцы csv файла, полученного экспортом электронной таблицы из приложения Libreoffice.Calc.
5. Сделать вывод о функциональном назначении языка и интерпретатора awk.

6.4. Содержание отчёта:

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.

7. ПРАКТИЧЕСКАЯ РАБОТА №7

Регулярные выражения как модель формального языка

7.1. Цель работы

Понять общность между формально-множественным описанием языка и регулярных выражениях (regex). Научиться формулировать языковые ограничения на языке регулярных выражений. Получить навыки написания преобразователей абстрактных языков.

7.2. Задание

Изучить примеры регулярных выражений и таблицы их не-терминальных компонентов (групп символов).

7.3. Ход работы

1. Загрузить компьютер в Unix-совместимую ОС (в ОС Windows активизировать запуск эмулятора `msys2`, `cygwin`).
2. Используя статью (<https://www.ibm.com/developerworks/ru/library/auregexp/index.html>) разобрать примеры использования утилиты `grep`, `sed`, `emacs`, `perl` для обработки стандартного многострочного текста.
3. Изучить особенности эксплуатации утилиты `pcretest`.
4. Изучить особенности эксплуатации утилиты `txt2regex`.
5. Проанализировать структуру подготовленных регулярных выражений.

6. Написать программу на perl, производящую сопоставление текста с шаблоном и выводящие элементы текстовой последовательности так как сделано на flex и bison на практическом занятии № 5.

7.4. Содержание отчёта:

В ходе выполнения работы нужно вести журнал активных действий и наблюдений. Журнал нужно включить в отчёт по работе. В отчёт также нужно включить критический анализ проведённой исследовательской деятельности и выводы по работе.