|  |  |
| --- | --- |
| **Изображение выглядит как текст, эмблема, герб, нашивка  Автоматически созданное описание** | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Робототехника и комплексная автоматизация (РК)

КАФЕДРА Системы автоматизированного проектирования (РК-6)

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**по курсу «Программное обеспечение систем автоматизированного проектирования»**

Студент Долженко Анастасия Тимофеевна

Группа РК6-32Б

Тип задания Лабораторная работа №8

Вариант A22(W)

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Долженко А.Т.**

*подпись, дата фамилия, и.о.*

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Князева С.В.**

*подпись, дата фамилия, и.о.*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Москва, 2023 г**.*

**Задание к лабораторной работе**

**Постановка задачи**

Разработать конечный автомат для распознавания во входном потоке текстовой информации слов из латинских букв, где гласные буквы расположены в алфавитном порядке. Все слова должны передаваться конечному автомату строками потока стандартного ввода. Результаты их распознавания должны отображаться через поток стандартного вывода. Программная реализация конечного автомата должна формироваться генератором YACC по правилам регулярной грамматики

**Требования к программе**

Программа предназначается для обработки двоичных наборов, поступающих программе через поток стандартного ввода.

1.1.  Реализация автомата должна выводить 3 вида диагностических сообщений в зависимости от корректности вводимой строки: сообщение о корректном вводе, об ошибочном вводе и о пустой строке.

1.2. Программа должна работать корректно в соответствии с заданием.

2. Требования к надежности

Программа должна работать с пустыми строками, а также со строками, содержащими символы помимо нулей и единиц. При вводе ошибочной или пустой строки программа должна выдать соответствующее диагностическое сообщение и продолжить работу.

3. Требования к составу и параметрам технических средств

Программа должна быть разработана исходя из возможности реализации на стандартном составе технических средств компьютеров любой архитектуры, после соответствующей трансляции исходного кода.

4. Условия эксплуатации

4.1. Программа реализации автомата должна быть ориентирована на эксплуатацию в среде OS UNIX

4.2. Программа должна быть реализована в виде выполняемого файла с именем lab8, по которому она должна вызываться средствами любого командного процессора OS UNIX.

4.3. Программа должна эксплуатироваться в режиме фильтра, который читает поток стандартного ввода и отображает результаты в потоке стандартного вывода OS UNIX по мере поступления информации.

5. Требования к информационной и программной совместимости

5.1. При разработке программы необходимо использовать генератор синтаксических анализаторов (далее по тексту - yacc) из состава инструментальных средств OS UNIX, который ориентирован на обработку файла спецификаций (далее по тексту - yacc-файл) проектируемого синтаксического анализатора.

5.2. При разработке программы необходимо составить yacc-файл, отражающий специфику лексического анализа кооментариев, и сохранить его под именем automaton.y в выбранном рабочем каталоге файловой системы OS UNIX.

5.3. Yacc-файл lab8.y программы реализации автомата должен содержать 3 раздела:

* раздел деклараций;
* раздел правил;
* раздел функций,

которые должны разделять символьные пары %%.

5.4. В разделе правил yacc-файла программы должны быть введены правила КС-грамматики, которые обеспечивают работу автомата.

5.5. Правила раздела правил yacc-файла программы должны обеспечивать выполнение следующих действий при распознавании лексем:

* Обработка корректного ввода
* Обработка ошибочного ввода
* Обработка ввода пустой строки

5.8. В разделе функций yacc-файла lab8.y программы реализации автомата необходимо специфицировать основную функцию main(), которая должна установить игнорирование клавиатурных сигналов, указанных в подразделе 2, и обеспечить вызов функции yyparse (), формируемой yacc по yacc-файлу для обработки формальной КС-грамматики, заявленной в разделе правил.

**Стадии и этапы разработки**

В процессе разработки программы автомата средствами yacc необходимо выполнить следующие этапы:

* подготовить файл спецификаций синтаксического анализатора (yacc-файл) для программы реализации автомата с именем lab8.y, используя любой текстовый редактор OS UNIX, например, xedit;
* получить исходный С-код программы в файле lab8.c, обработав yacc-файл lab8.y командой yacc следующим образом:

$ yacc lab8.y -o lab8.c

* построить по исходному коду lab8.c выполняемый файл lac программы, вызвав C-компилятор OS UNIX:

$ cc -o lab8 lab8.c -ll

**Код программы**

%{

#include<stdio.h>

int yylex();

int main();

void yyerror();

%}

%start input

%%

input: input '\n' {printf("Correct\n");}

|input aa '\n' {printf("Correct\n");};

|input ee '\n' {printf("Correct\n");}

|input ii '\n' {printf("Correct\n");}

|input oo '\n' {printf("Correct\n");}

|input uu '\n' {printf("Correct\n");}

|input yy '\n' {printf("Correct\n");}

|'\n' { printf("Empty line\n"); }

|error '\n' { yyerrok; };

aa: 'a' | 'A' | aa 'a'| aa 'A'| aa cons | cons aa;

ee: 'e' | 'E' | ee 'e'| ee 'E'| ee cons | cons ee | aa ee;

ii: 'i' | 'I' | ii 'i'| ii 'I'| ii cons | cons ii | aa ii | ee ii;

oo: 'o' | 'O' | oo 'o'| oo 'O'| oo cons | cons oo | aa oo | ee oo | ii oo;

uu: 'u' | 'U' | uu 'u'| uu 'U'| uu cons | cons uu | aa uu | ee uu | ii uu | oo uu;

yy: 'y' | 'Y' | yy 'y'| yy 'Y'| yy cons | cons yy | aa yy | ee yy | ii yy | oo yy | uu yy;

cons: 'b'|'c'|'d'|'f'|'g'|'h'|'j'|'k'|'l'|'m'

|'n'|'p'|'q'|'r'|'s'|'t'|'v'|'w'|'x'|'z'

|'B'|'C'|'D'|'F'|'G'|'H'|'J'|'K'|'L'|'M'

|'N'|'P'|'Q'|'R'|'S'|'T'|'V'|'W'|'X'|'Z'

|cons 'b'|cons 'c'|cons 'd'|cons 'f'|cons 'g'

|cons 'h'|cons 'j'|cons 'k'|cons 'l'|cons 'm'

|cons 'n'|cons 'p'|cons 'q'|cons 'r'|cons 's'

|cons 't'|cons 'v'|cons 'w'|cons 'x'|cons 'z'

|cons 'B'|cons 'C'|cons 'D'|cons 'F'|cons 'G'

|cons 'H'|cons 'J'|cons 'K'|cons 'L'|cons 'M'

|cons 'N'|cons 'P'|cons 'Q'|cons 'R'|cons 'S'

|cons 'T'|cons 'V'|cons 'W'|cons 'X'|cons 'Z';

%%

int yylex(){return(getchar());}

void yyerror() { printf("Incorrect\n");}

int main(){printf("Enter the line:\n"); return (yyparse());}

**Результаты работы программы**

**Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание**

**Список литературы**

1. Рейуорд-Смит В.Дж. Теория формальных языков. Вводный курс, M.: Радио и связь, 1988.  
2. Тихомиров В.П., Давидов М.И. Операционная система ДЕМОС: инструментальные средства программирования, М.: Финансы и статистика, 1988. 3. SCO XENIX, Development System, YACC Programmer Guide, SCO Inc., 1986.

4. Конспекты лекций 5. http://bigor.bmstu.ru