|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Мытищинский филиал**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Космический

КАФЕДРА «Прикладная математика, информатика и вычислительная техника» К3-МФ

**Лабораторная работа**

*ПО ДИСЦИПЛИНЕ:*

***Системное программное обеспечение***

***НА ТЕМУ:***

***\_\_\_\_\_\_\_\_Совместное использование YACC и LEX\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_при помощи утилиты Make \_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

Студент \_\_К3-53Б\_\_  **\_\_\_\_\_\_\_\_\_\_** Цветков Юрий Алексеевич

(Группа) (Подпись, дата) (И.О.Фамилия)

#### Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_ Чернышов Александр Викторович

(Подпись, дата) (И.О.Фамилия)

*2021 г.*

**Задание**

Используя калькулятор из 1 лабораторной, написать лексический разбор входного потока с помощью LEX и реализовать их совместную сборку при помощи утилиты Make.

***Код calc.y***

%{

#define YYSTYPE double

%}

%token DATA

%left ';'

%%

list:

| list '\n'

| list wyrag '\n' {printf("result: %g\n", $2);};

wyrag : DATA {$$=$1;}

| wyrag ';' wyrag {

int n = $1, m = $3;

if(n < 0 || m < 0 || m > n)

{

printf("error number!!!\n");

return;

}

$$ = FactTree(n) / (FactTree(m) \* FactTree(n - m));

};

%%

#include <stdio.h>

#include <ctype.h>

int ProdTree(int l, int r)

{

if(l > r)

return 1;

if(l == r)

return l;

if(r - l == 1)

return l \* r;

int m = (l + r) / 2;

return ProdTree(l, m) \* ProdTree(m + 1, r);

}

int FactTree(int n)

{

if(n < 0)

return 0;

if(n == 0)

return 1;

if(n == 1 || n == 2)

return n;

return ProdTree(2, n);

}

main()

{

return yyparse();

}

yyerror(s)

char\*s;

{

fprintf(stderr, "%s\n", s);

}

//Функция ниже будет заменена программой лексического разбора на LEX

/\*yylex()

{

int c;

do{

c=getchar();

}while(c==' ' || c == '\t');

if(c==EOF)

return 0;

if(isdigit(c)){

ungetc(c, stdin);

scanf("%lf", &yylval);

return DATA;

}

return c;

}\*/

***Код calc.l***

%{

#define YYSTYPE double

#include "y.tab.h"

extern YYSTYPE yylval;

%}

%%

[0-9]+ {

yylval = atof(yytext);

return DATA;

}

[ \t]+ ; //пропускаем пробелы и табуляции

\n { return yytext[0]; }; // для вывода результата

. { return yytext[0]; }; // для операций ( . - все символы, которые не попали ни под одно правило)

%%

int yywrap()

{

return 1;

}

***Код makefile***

l = lex.yy.c  
y = [y.tab](https://vk.com/away.php?to=http%3A%2F%2Fy.tab&cc_key=" \t "_blank).c [y.tab](https://vk.com/away.php?to=http%3A%2F%2Fy.tab&cc_key=" \t "_blank).h  
prog = calc  
  
$(prog):$(l) $(y)  
 gcc $(l) $(y) -o $(prog)  
 rm $(l) $(y)  
  
$(yacc): $(prog).y  
 yacc -d $(prog).y  
  
$(lex): $(prog).l $(y)  
 flex $(prog).l

**Сборка и запуск программы**

**Make -** данная команда сама соберёт программу, на основе кода, написанного в **makefile**

**./calc** – запуск программы

**Результат работы программы**

После запуска программы убеждаемся, что всё работает корректно и результат соответствует действительности

