**ASSIGNMENT NO.**

NAME: ROHIT ANIL GUND

CLASS: BE COMP-1 ROLL NO.: 402055

**PROGRAM:**

%{

#include <stdio.h>

#include<string.h>

#include<stdlib.h>

struct LList

{

char nm[10];

char type[10];

struct LList \*next;

};

struct LList \*head;

void insert1(char \*,char\*);

void display();

int search(char \*);

%}

letter [a-zA-Z]

digit [0-9]

op "="|"+"|"-"|"\*"|"/"

conditionalop "=="|"<"|">"|"<="|">="

datatype "int"|"float"|"char"

variable {letter}({letter}{digit})\*

ignore " "|";"|"("|")"|","|"\""

parenthesis "("|")"|"{"|"}"|"["|"]"

%%

<<EOF>> {return 0;}

{parenthesis} {printf("parentesis :- %s\n",yytext);}

"main"|"getch()"|"sizeof"/"("{variable}")" {printf("System Function :- %s\n",yytext);}

"if"/"("{variable}{conditionalop}{variable}")" {printf("conditional statement :- %s\n",yytext);}

"while"/"("{variable}{conditionalop}{variable}")" {printf("loop statement :- %s\n",yytext);}

"printf(\""({letter}|{ignore}|{digit})\*"\")" {printf ("Inbuild Function :- %s\n",yytext);}

"getch"/"()" {printf("Inbuild Function :- %s\n",yytext);}

"#include <"({letter})+".h>" {printf("preprocessor directive :- %s\n",yytext);}

{datatype} {printf("datatype :- %s\n",yytext);}

{variable}+/"("({variable}(","{variable})\*)\*")" { insert1(yytext,"udfun");printf("userdefined function :- %s\n",yytext);}

{variable}+/"["({digit})+"]" {insert1(yytext,"Array");;printf("Array :- %s\n",yytext);}

"\*"{variable}+ {insert1(yytext,"pointer");printf("pointer :- %s\n",yytext);}

{variable} {insert1(yytext,"VAR");printf("userdefined variable :- %s\n",yytext);}

{op} {printf("operators :- %s\n",yytext);}

{ignore} {}

{conditionalop} {printf("Conditional Operator :- %s\n",yytext);}

"//"({letter}|{digit})+ {printf("comment :- %s\n",yytext);}

%%

int main()

{

yyin = fopen("temp.txt","r");

head=NULL;

yylex();

printf("\nSymbol Table here\n");

display();

yywrap();

}

struct LList \* newnode()

{

struct LList \*a=(struct LList \*)malloc(sizeof(struct LList));

a->next=NULL;

strcpy(a->nm,"");

strcpy(a->type,"");

return a;

}

void insert1(char \*anm,char \*atype)

{

if(head==NULL)

{

head=newnode();

strcpy(head->nm,anm);

strcpy(head->type,atype);

return;

}

if(search(anm)==0)

{

struct LList \*a=newnode();

a=head->next;

if(head->next==NULL)

{

head->next=newnode();

strcpy((head->next)->nm,anm);

strcpy((head->next)->type,atype);

}

else

{

while(a->next!=NULL)

{

a=a->next;

}

a->next=newnode();

strcpy((a->next)->nm,anm);

strcpy((a->next)->type,atype);

}

}

}

int search(char \*anm)

{

struct LList \*a=newnode();

a=head->next;

while(a)

{

if(strcmp(a->nm,anm)==0)

break;

a=a->next;

}

if(a!=NULL)

return 1;

else

return 0;

}

void display()

{

if(head==NULL)

{

printf("\nEmpty Symbol Table");

return;

}

struct LList \*a;

a=newnode();

a=head->next;

while(a)

{

printf("\t%s\t%s\n",a->nm,a->type);

a=a->next;

}

}

int yywrap()

{

return 1;

}

**INPUT :**

#include <iostream.h>

int main()

{

char b = 'a';

char \*c = b;

int a = sizeof(char),d = 0;

while(a>d)

{

if(d<a)

{

a = a + now();

printf("Student");

}

}

getch();

}

**OUTPUT:**



