***Experiment No.***

1. ***A.I.M-*** Design a Lexical analyzer for identifying different types of token used in C language.

Source Code -

#include<stdio.h>

#include<conio.h>

#include<string.h>

int key=0;

char expr[100];

char cont[][20]={"CONTROLS","for","do","while","NULL"};

char cond[][20]={"CONDITIONS","if","else","elseif","NULL"};

char optr[][20]={"OPERATOR","+","-","\*","/","%","<",">","<=",">=","=","(",")","NULL"};

char branch[][20]={"BRANCHING","goto","jump","NULL"};

void checking(char[],char[][20]);

int main()

{

int i,j,k,l,m,n,c,c1,a;

char sbexpr[50],txt[3];

printf("enter the expression: ");

gets(expr);

for(i=0;expr[i]!=NULL;i++)

{

key=0;

for(j=i,k=0;expr[i]!=32 && expr[i]!=NULL;i++,j++,k++)

sbexpr[k]=expr[j];

sbexpr[k]=NULL;

if(key==0)

checking(sbexpr,cond);

if(key==0)

checking(sbexpr,cont);

if(key==0)

checking(sbexpr,branch);

if(key==0)

checking(sbexpr,optr);

if(key==0)

{

c=0;c1=0;

for(m=0;sbexpr[m]!=NULL;m++)

{

key=0;

if((key==0) || ((sbexpr[m]>97 && sbexpr[m]<=122) || (sbexpr[m]>=65 && sbexpr[m]<=90)|| (sbexpr[m]>=48 && sbexpr[m]<=57)))

{

c++;

if(sbexpr[m]>=48 && sbexpr[m]<=57)

c1++;

key=1;

}

}

if(c1==c)

printf("%s\t--------> Identifier(It is a numerical value)\n",sbexpr);

else if(m==c)

printf("%s\t--------> Identifier\n",sbexpr);

else

printf("%s\t--------> Invalid Symbol\n",sbexpr);

key=1;

}

if(key==0)

{

printf("%s\t-----------> Address\n",sbexpr);

key=1;

}

}

printf("if you want to continue press 1 otherwise 0 ");

scanf("%d",&a);

if(a=1)

main();

else

printf(" Program is over ");

getch();

}

void checking(char expr[],char check[][20])

{

int i;

for(i=1;strcmp(check[i],"NULL")!=0;i++)

{

if(strcmp(expr,check[i])==0)

{

printf("%s\t--------> %s\n",expr,check[0]);

key=1;

break;

}

}

}

Output -:

