Ex No: 1 **Date:**

IMPLEMENT CODE TO RECOGNIZE TOKENS IN C

AIM:

To implement the program to identify C keywords, identifiers, operators, end statements like [], {} using C tool.

ALGORITHM:

Step 1 :Initialize a list or array to store the address code sequence.

Step 2 :Retrieve the current location of operand 3 using its address.

Step 3 :Generate a move instruction to move the operand to the desired location (B, O).

Step 4 :Update the address of operand A.

Step 5 :If the current values of B and () are null, exit the process.

Step 6 :If B and () are not null, generate the operator () A, 3 ADPR instruction.

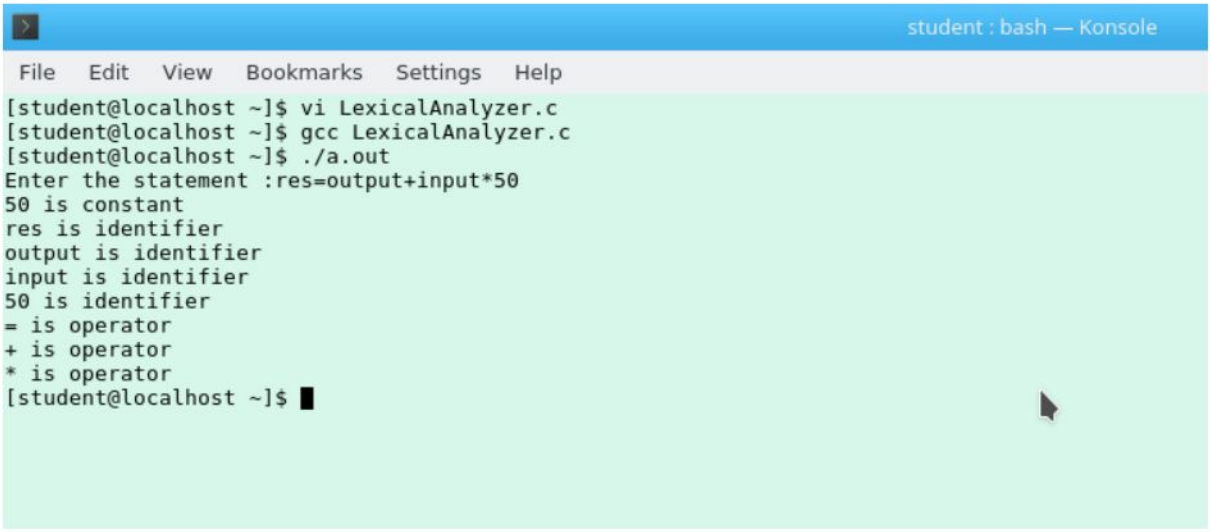
step 7 :Store the generated move instruction in memory.

PROGRAM:

```
#include<stdio.h>
#include<string.h>
int main()
{
 char  str[100]={0};
 printf("Enter the statement :");
 scanf("%s",&str);


char str2[100];
strcpy(str2,str);

const char
chara[100]={'a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v','w','x','y','z','A','B','C','D','E','F','
G',
'H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z',' ','=','+','-','*','/'};
char *token3;
token3=strtok(str2,chara);
while(token3!=NULL)
{
printf("%s is constant\n",token3);
token3=strtok(NULL,chara);
}

const char deli[50]={' ','=','+','-','*','/'};
```

```
char *token;
token = strtok(str,deli);
while(token!=NULL)
{
printf("%s is identifier\n",token);
token=strtok(NULL,deli);
}
const char
alpha[100]={'a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v','w','x','y','z','A','B','C','D','E','F','
G',
'H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','0','1','2','3','4','5','6','7','8','9'};

char *token2;
token2=strtok(str2,alpha);
while(token2!=NULL)
{
printf("%s is operator\n",token2);
token2=strtok(NULL,alpha);
}

 return 0;
}
```

OUTPUT:



RESULT