AIM

To implement two pass Macro Processor in C language.

ALGORITHM

Step 1 : START

Step 2 : Include necessary headerfiles.

Step 3 : Declare a structure macro with members opc[], oper[].and label[] and structure d.

Step 4 : Declare arrays for label, opcode, macro name operand, new label and new operand

Step 5 : Declare necessary variables and file pointers.

Step 6 : Open file MACIN.DAT in read mode and MACOUT.DAT and DEFTAB.DAT in write mode.

Step 7 : Read the input file (first line).

Step 8 : Do the following till Opcode = END.

Step 8.1 : If the opcode is MACRO perform the following

Step 8.1.1 : Copy the macroname and label.

Step 8.1.2 : Read the next line of input line

Step 8.1.3 : Set lines = 0.

Step 8.1.4 : Perform the following till the opcode is MEND

Step 8.1.4.1 : Write the label operand and opcode to deftab file.

Step 8.1.4.2 : Copy the corresponding opcode, operand and label.

Step 8.1.4.3 : Read the next input line.

Step 8.1.4.4 : lines ++ and go to Step 8.1.4.

Step 8.2 : If the opcode and macroname are the same perform the following

Step 8.2.1 : Write the label, opcode, operand values of all the lines into output file.

Step 8.3 : Otherwise

Step 8.3.1 : Write the label, opcode, operand to output file.

Step 8.3.2 : Read the next line of opw file.

Step 9 : Write the remaining label, opcode and its operand to output file.

Step 10 : Close all the files.

Step 11 : Display completed message

Step 12 : STOP.

PROGRAM

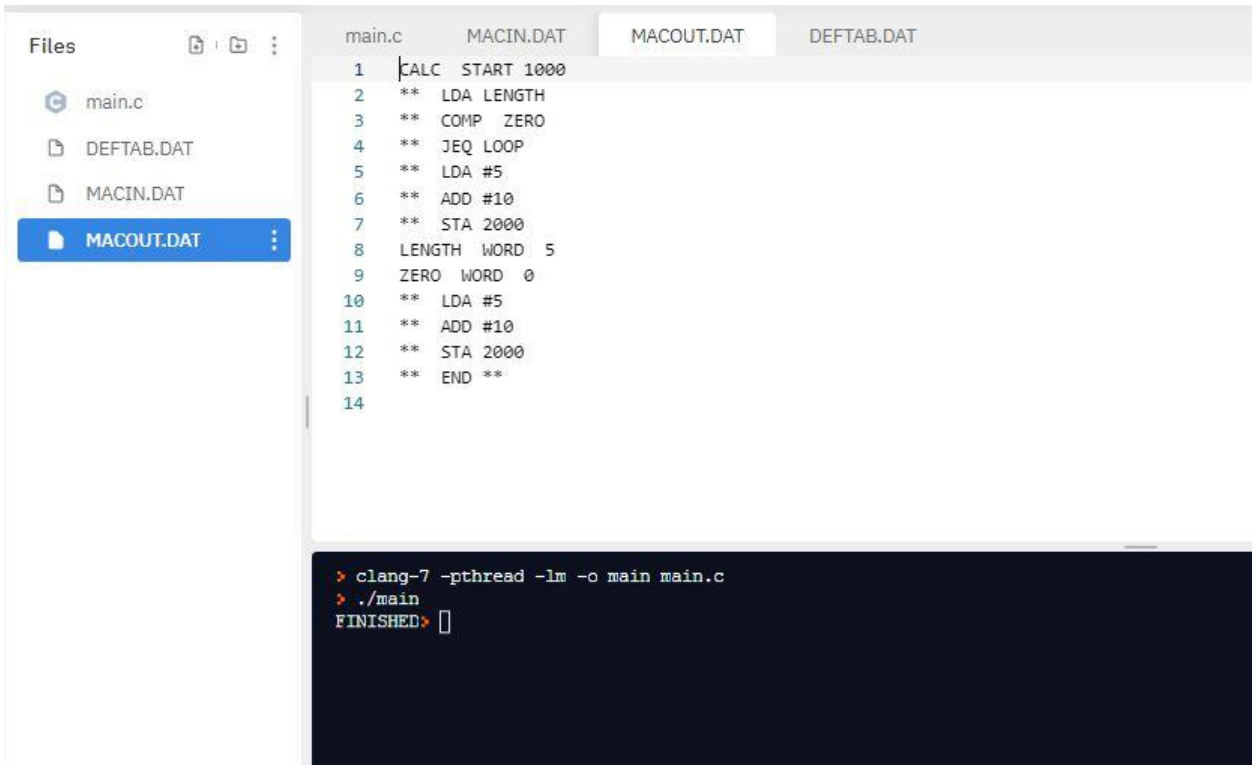
```
#include<stdio.h>
#include<conio.h>
#include<string.h>
struct macro
{
 char lab[10];
 char opc[10];
 char oper[10];
}d[10];
void main()
{
 char label[10],opcode[10],operand[10],newlabel[10],newoperand[10];
 char macroname[10];
 int i,lines;
 FILE *f1,*f2,*f3;
 f1 = fopen("MACIN.DAT","r");
 f2 = fopen("MACOUT.DAT","w");
 f3 = fopen("DEFTAB.DAT","w");
 fscanf(f1,"%s %s %s",label,opcode,operand);
 while(strcmp(opcode,"END")!=0)
 {
  if(strcmp(opcode,"MACRO")==0)
  {
    strcpy(macroname,label);
    fscanf(f1,"%s%s%s",label,opcode,operand);
    lines = 0;
    while(strcmp(opcode,"MEND")!=0)
    {
      fprintf(f3,"%s\t%s\t%s\n",label,opcode,operand);
      strcpy(d[lines].lab,label);
      strcpy(d[lines].opc,opcode);
      strcpy(d[lines].oper,operand);
      fscanf(f1,"%s %s %s",label,opcode,operand);
      lines++;
```

```
   }
  }
  else if(strcmp(opcode,macroname)==0)
  {
    for(i=0;i<lines;i++)
    {
      fprintf(f2,"%s\t%s\t%s\n",d[i].lab,d[i].opc,d[i].oper);
    }
  }
  else
    fprintf(f2,"%s\t%s\t%s\n",label,opcode,operand);
    fscanf(f1,"%s%s%s",label,opcode,operand);
 }
 fprintf(f2,"%s\t%s\t%s\n",label,opcode,operand);
 fclose(f1);
 fclose(f2);
 fclose(f3);
 printf("FINISHED");
}
```

OUTPUT

Input

Output


Files
main.c
DEFTAB.DAT
MACIN.DAT
MACOUT.DAT
main.c
MACIN.DAT
MACOUT.DAT
DEFTAB.DAT
1 CALC START 1000
2 ** LDA LENGTH
3 ** COMP ZERO
4 ** JEQ LOOP
5 ** LDA #5
6 ** ADD #10
7 ** STA 2000
8 LENGTH WORD 5
9 ZERO WORD 0
10 ** LDA #5
11 ** ADD #10
12 ** STA 2000
13 ** END **
14
> clang-7 -pthread -lm -o main main.c
> ./main
FINISHED>

RESULT

Single pass macroprocessor alternates between macro definition and macro expansion.Such a macroprocessor is implemented using C program. The output files are also displayed.

Submitted By

SAGAR SAJI

S5 CSE-B

37