Program

//PASS 1 OF TWO PASS ASSEMBLER

```
#include<stdio.h>
#include<string.h>
#include <stdlib.h>
void main(){
   char opcode[10], operand[10], label[10], mnemonic[10], code[10];
   int locctr, start, length;
   FILE *input, *optab, *symbol, *output;
   input = fopen("input.txt", "r");
   optab = fopen("optab.txt", "r");
   symbol = fopen("symbol.txt", "w");
   output = fopen("output.txt", "w");
   fscanf(input,"%s\t%s\t%s",label,opcode,operand);
   if(strcmp(opcode,"START")==0){
      start = atoi(operand);
      locctr = start;
      fprintf(output, "\t%s\t%s\t%s\n",label,opcode,operand);
      fscanf(input,"%s\t%s\t%s",label,opcode,operand);
   } else {
      locctr = 0;
   }
   while(strcmp(opcode,"END")!=0){
      fprintf(output, "%d\t",locctr);
      if(strcmp(label,"-")!=0){
         fprintf(symbol, "%s\t%d\n",label,locctr);
      }
      fscanf(optab,"%s\t%s",code,mnemonic);
      while(strcmp(code,"END")!=0){
         if(strcmp(opcode,code)==0){
            locctr += 3;
            break;
         }
         fscanf(optab,"%s\t%s",code,mnemonic);
      }
      if(strcmp(opcode,"WORD")==0){
         locctr += 3;
      }
      else if(strcmp(opcode,"RESW")==0){
         locctr += (3*(atoi(operand)));
      }
      else if(strcmp(opcode,"RESB")==0){
         locctr += atoi(operand);
      }
      else if(strcmp(opcode,"BYTE")==0){
         locctr+=strlen(operand)-3;
      }
      fprintf(output, "%s\t%s\t%s\t\n",label,opcode,operand);
```

```
        fscanf(input,"%s\t%s\t%s",label,opcode,operand);
    }
    fprintf(output, "%d",locctr);
    fprintf(output, "\t%s\t%s\t%s\n",label,opcode,operand);
    length = locctr-start;
    printf("The length of code: %d\n",length);
    fclose(input);
    fclose(optab);
}
```

//OPTAB.txt

```
START   *
LDA     *
STA     *
LDCH    *
STCH    *
END     *
```

//INPUT.txt

```
COPY    START   2000
**      LDA     FIVE
**      STA     ALPHA
**      LDCH    CHARZ
**      STCH    C1
ALPHA   RESW    1
FIVE    WORD    05
CHARZ   BYTE    c'EOF'
C1      RESB    1
**      END     **
```

Program

//PASS 2 OF TWO PASS ASSEMBLER

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main()
{
char a[10],ad[10],label[10],opcode[10],operand[10],symbol[10],ch;
int st,diff,i,address=0,add,len,actual_len,finaddr,prevaddr,j=0;
char mnemonic[15][15]={"LDA","STA","LDCH","STCH", "END"};
char code[15][15]={"0","20","30","40", "00"};
FILE *fp1,*fp2,*fp3,*fp4;
fp1=fopen("out.txt","w");
fp2=fopen("symtab.txt","r");
fp3=fopen("input2.txt","r");
fp4=fopen("obcode.txt","w");
fscanf(fp3,"%s%s%s",label,opcode,operand);
while(strcmp(opcode,"END")!=0)
{
prevaddr=address;
fscanf(fp3,"%d%s%s%s",&address,label,opcode,operand);
}
finaddr=address-2000;
fclose(fp3);
fp3=fopen("input2.txt","r");

fscanf(fp3,"%s%s%s",label,opcode,operand);
if(strcmp(opcode,"START")==0)
{
fprintf(fp1,"\t%s\t%s\t%s\n",label,opcode,operand);
fprintf(fp4,"H^%s^00%s^00%d\n",label,operand,finaddr);
fscanf(fp3,"%d%s%s%s",&address,label,opcode,operand);
st=address;
diff=prevaddr-st;
fprintf(fp4,"T^00%d^%d",address,diff);
}
while(strcmp(opcode,"END")!=0)
{
if(strcmp(opcode,"BYTE")==0)
{
fprintf(fp1,"%d\t%s\t%s\t%s\t",address,label,opcode,operand);
len=strlen(operand);
actual_len=len-3;
fprintf(fp4,"^");
for(i=2;i<(actual_len+2);i++)
{
sprintf(ad, "%x", operand[i]);
```

```
fprintf(fp1,"%s",ad);
fprintf(fp4,"%s",ad);
}
fprintf(fp1,"\n");
}
else if(strcmp(opcode,"WORD")==0)
{
len=strlen(operand);
sprintf(a,"%x",atoi(operand));
fprintf(fp1,"%d\t%s\t%s\t%s\t00000%s\n",address,label,opcode,operand,a);
fprintf(fp4,"^00000%s",a);
}
else if((strcmp(opcode,"RESB")==0)||(strcmp(opcode,"RESW")==0))
fprintf(fp1,"%d\t%s\t%s\t%s\n",address,label,opcode,operand);
else
{
while(strcmp(opcode,mnemonic[j])!=0)
j++;
if(strcmp(operand,"COPY")==0)
fprintf(fp1,"%d\t%s\t%s\t%s\t%s0000\n",address,label,opcode,operand,code[j]);
else
{
rewind(fp2);
fscanf(fp2,"%s%d",symbol,&add);
while(strcmp(operand,symbol)!=0)
fscanf(fp2,"%s%d",symbol,&add);
fprintf(fp1,"%d\t%s\t%s\t%s\t%s%d\n",address,label,opcode,operand,code[j],add);
fprintf(fp4,"^%s%d",code[j],add);
}
}
fscanf(fp3,"%d%s%s%s",&address,label,opcode,operand);
}
fprintf(fp1,"%d\t%s\t%s\t%s\n",address,label,opcode,operand);
fprintf(fp4,"\nE^00%d",st);
printf("\n Intermediate file is converted into object code");
fclose(fp1);
fclose(fp2);
fclose(fp3);
fclose(fp4);

printf("\n\nThe contents of Intermediate file:\n\n\t");
fp3=fopen("input2.txt","r");
ch=fgetc(fp3);
while(ch!=EOF)
{
printf("%c",ch);
ch=fgetc(fp3);
}
printf("\n\nThe contents of Symbol Table :\n\n");
fp2=fopen("symtab.txt","r");
```

```
ch=fgetc(fp2);
while(ch!=EOF)
{
printf("%c",ch);
ch=fgetc(fp2);
}
printf("\n\nThe contents of Output file :\n\n");
fp1=fopen("out.txt","r");
ch=fgetc(fp1);
while(ch!=EOF)
{
printf("%c",ch);
ch=fgetc(fp1);
}
printf("\n\nThe contents of Object code file :\n\n");
fp4=fopen("obcode.txt","r");
ch=fgetc(fp4);
while(ch!=EOF)
{
printf("%c",ch);
ch=fgetc(fp4);
}
fclose(fp1);
fclose(fp2);
fclose(fp3);
fclose(fp4);
printf("\n");
}
```

//INPUT.txt

```
        COPY    START   2000
2000    **      LDA     FIVE
2003    **      STA     ALPHA
2006    **      LDCH    CHARZ
2009    **      STCH    C1
2012    ALPHA   RESW    1
2015    FIVE    WORD    05
2018    CHARZ   BYTE    c'EOF'
2021    C1      RESB    1
2022    **      END     **
```

//SYMTAB.txt

```
ALPHA   2012
FIVE    2015
CHARZ   2018
C1      2021
```

//OPTAB.txt

```
LDA     00
STA     20
LDCH    30
STCH    40
END     00
```

Program

//ABSOLUTE LOADER

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<stdlib.h>
void main()
{
  FILE *fp;
  int i,addr1,l,j,staddr1;
  char name[10],line[50],name1[10],addr[10],rec[10],ch,staddr[10];
  clrscr();
  printf("enter program name:" );
  scanf("%s",name);
  fp=fopen("abssrc.txt","r");
  fscanf(fp,"%s",line);
  for(i=2,j=0;i<8,j<6;i++,j++)
    name1[j]=line[i];
    name1[j]='\0';
  printf("name from obj. %s\n",name1);
  if(strcmp(name,name1)==0)
   {
    do
    {
    fscanf(fp,"%s",line);
    if(line[0]=='T')
    {
    for(i=2,j=0;i<8,j<6;i++,j++)
    staddr[j]=line[i];
    staddr[j]='\0';
    staddr1=atoi(staddr);
    i=12;
    while(line[i]!='$')
    {
      if(line[i]!='^')
      {
        printf("00%d \t %c%c\n", staddr1,line[i],line[i+1]);
        staddr1++;
        i=i+2;
      }
```

```
      else i++;
    }
    }
    else if(line[0]='E')
    fclose(fp);
    }while(!feof(fp));
    }

getch();
}
```

//INPUT (abssrc.txt)

```
H^SAMPLE^001000^0035
T^001000^0C^001003^071009$
T^002000^03^111111$
E^001000
```

//ADD TWO 16 BIT NUMBERS

```
ASSUME CS:CODE,DS:DATA
DATA SEGMENT
MSG1 DB 0AH, 'Enter the first number: $'
MSG2 DB 0AH, 'Enter the second number: $'
MSG3 DB 0AH, 'The sum is: $'
DATA ENDS
CODE SEGMENT
START:
MOV AX,DATA
MOV DS,AX
MOV DX,OFFSET MSG1
MOV AH,09H
INT 21H
MOV AH,01H
INT 21H
SUB AL,30H
MOV BH,0AH
MUL BH
MOV BH,AL
MOV AH,01H
INT 21H
SUB AL,30H
ADD BH,AL
MOV AH,01H
INT 21H
SUB AL,30H
MOV CH,0AH
MUL CH
MOV CH,AL
MOV AH,01H
INT 21H
SUB AL,30H
ADD CH,AL

MOV DX,OFFSET MSG2
MOV AH,09H
INT 21H
MOV AH,01H
INT 21H
SUB AL,30H
MOV BL,0AH
MUL BL
MOV BL,AL
MOV AH,01H
INT 21H
SUB AL,30H
ADD BL,AL
MOV AH,01H
```

```
INT 21H
SUB AL,30H
MOV DH,0AH
MUL DH
MOV DH,AL
MOV AH,01H
INT 21H
SUB AL,30H
ADD DH,AL
ADD BH,BL
ADD CH,DH

MOV DX,OFFSET MSG3
MOV AH,09H
INT 21H
MOV CL,0AH
MOV AL,BH
MOV AH,00H
DIV CL
MOV BL,AH
ADD AL,30H
MOV DL,AL
MOV AH,02H
INT 21H
ADD BL,30H
MOV DL,BL
MOV AH,02H
INT 21H
MOV CL,0AH
MOV AL,CH
MOV AH,00H
DIV CL
MOV BL,AH
ADD AL,30H
MOV DL,AL
MOV AH,02H
INT 21H
ADD BL,30H
MOV DL,BL
MOV AH,02H
INT 21H
MOV AH,4CH
INT 21H

CODE ENDS
END START
```

//REVERSE A STRING

```
.model small
.stack 100H
.data

MSG1 DB 13,10, "Enter String : $"
MSG2 DB 13,10, "Reverse String is : $"
.code

MOV AX, @DATA
MOV DS, AX
LEA DX, MSG1
MOV AH, 09H
INT 21H
MOV CX,00H
READ:
MOV AH, 01
INT 21H
CMP AL, 0DH
JE AHEAD
PUSH AX
INC CX
JMP READ
AHEAD:
LEA DX, MSG2
MOV AH,09H
INT 21H
DISPLAY:
MOV AH,02H
POP DX
INT 21H
LOOP DISPLAY
MOV AH,4CH
INT 21H
END
```