



SOMAIYA
VIDYAVIHAR UNIVERSITY

K J Somaiya College of Engineering



K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Batch: A2

Roll No.: 1911027

Experiment No. 10

Grade: AA / AB / BB / BC / CC / CD / DD

Signature of the Staff In-charge with date

TITLE: Design and implement 2 pass Macro Processor

AIM: To learn working of two pass macro preprocessor and databases required for implementation of Macro Processor.

Expected Outcome of Experiment:

CO 5. To understand the designing and implementation of system software like Assembler, Macro preprocessor and linker loader

Books/ Journals/ Websites referred:

1. Jhon J. Donovan : Systems programming, Tata McGraw Hill

Pre Lab/ Prior Concepts:

How macros are used in assembly language Program.

Algorithm for PASS – I

1. Set MDTC(macro definition table counter) to 1
2. Set MNTC(macro name table counter) to 1
3. Read next statement from source program

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

4. If this source statement is pseudo-opcode MACRO (start of macro definition) then goto step 5 else goto step 16
5. Read next statement from source program(macro name line)
6. Enter macro name found in step 5 in name field of MNT(macro name table) and Also enter current value of MDTC in MDT index field of MNT
7. Increment MNTC by 1
8. Prepare argument list array
9. Enter macro name into MDT at index MDTC
10. Increment MDTC by one
11. Read next statement from source program.
12. Create and substitute index notation for arguments in the source statement if any
13. Enter this line into the MDT
14. Increment MDTC by one
15. Check if currently read source statement is pseudo-opcode MEND. If yes then goto step 3 else goto step 11
16. Write source program statement as it is in file(output of pass 1 s.p.)
17. Check if pseudo opcode END(end of source program) is encountered . If yes goto step 18 else goto step 19
18. goto PASS – II
19. Go to step 3
20. end of PASS –I

Algorithm for PASS – II

1. Read next statement from source program(from output s.p. file of pass I)
2. Search in MNT for match with operation code(mnemonic)
3. If macro name found the goto step 4 else goto step 11
4. Retrieve MDT index from MNT and store it in MDTP(macro definition table pointer).
5. Set up argument list array (store actual parameter in ala)
6. Increment MDTP by 1
7. Retrieve line pointer by MDTP from MDT
8. Substitute index notation by actual parameter from argument list array if any
9. Check if currently retrieved line is pseudo opcode MEND, if yes go to step 1 else goto step 10
10. Write the statement formed in step 8 to expanded source file and go to step 6
11. Write source statement directly into expanded source file

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

12. Check if pseudo opcode END(end of s.p. output of pass I) encountered, if yes goto step 13 else goto step 1
13. End of PASS-II

Stepwise-Procedure:

- a. Read a Source Program as input.
- b. Recognise the Macro Definition and
- c. Create Macro Name Table (MNT), and Macro Definition Table (MDT), and Prepare ALA.
- d. Recognise the Macro Call statement
- e. Update the Macro Definition Table Pointer(MDTP), and Prepare ALA.
- f. Expand the macro call by reading instructions from the MDT.

Sample input and output:

Input file:

```
MACRO
INCR1 &FIRST,&SECOND=DATA9
A    1,&FIRST
L    2,&SECOND
MEND
MACRO
INCR2 &ARG1,&ARG2
L    3,&ARG1
ST   4,&ARG2
MEND
PRG2 START
      USING      *,BASE
INCR1 DATA1,DATA2
INCR2 DATA3,DATA4
FOUR DC    F'4'
FIVE  DC    F'5'
BASE EQU    8
TEMP  DS    '1'F
      DROP 8
      END
```

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Sample Output:

Intermediate source file generated by Pass-I

```
PRG2 START
      USING      *,BASE
INCR1 DATA1,DATA12
INCR2 DATA3,DATA4
FOUR DC    F'4'
FIVE  DC    F'5'
BASE EQU   8
TEMP DS    '1'F
      DROP 8
      END
```

Macro Name Table (MNT) created by Pass-1

INDEX	MACRO NAME	MDT INDEX
1	INCR1	1
2	INCR2	5

Argument List Array (ALA) created by Pass-1 for INCR1 Macro

INDEX	ARGUMENT
1	&FIRST
2	&SECOND

Argument List Array (ALA) created by Pass-1 for INCR2 Macro

INDEX	ARGUMENT
1	&ARG1
2	&ARG2

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Macro Definition table (MDT) created by Pass-1

INDEX	DEFINITION
1	INCR1 &FIRST,&SECOND
2	A 1,#1
3	L 2,#2
4	MEND
5	INCR2 &ARG1,&ARG2
6	L 3,#1
7	ST 4,#2
8	MEND

Output generated by Pass-II

Argument List Array (ALA) created for INCR1 Macro call

INDEX	ARGUMENT
1	DATA1
2	DATA12

Argument List Array (ALA) created for INCR2 Macro call

INDEX	ARGUMENT
1	DATA3
2	DATA4

Expanded Source code generated by Pass-2

```

PRG2 START
      USING      *,BASE
      A 1,DATA1
      L 2,DATA12
      L 3,DATA3
      ST 4,DATA4
  
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
FOUR DC    F'4'  
FIVE  DC    F'5'  
BASE EQU   8  
TEMP DS    '1'F  
          DROP 8  
          END
```

Implementation details: (printout of code)

Code:

```
from prettytable import PrettyTable  
program=open('macro_program.txt','r')  
mac_pass1=open('macro_pass1.txt','a')  
mac_pass2=open('macro_pass2.txt','a')  
mdt=open('mdt.txt','a')  
mnt=open('mnt.txt','a')  
funcs=[]  
p_f=[]  
index=1  
mname_index=1  
for i in program:  
    p_f.append(i.split())  
macros=[]  
mname=[]  
i=0  
print("-----  
-----")  
print("-----2 PASS MACROS PROCESSOR-----  
-----")  
print("-----  
-----")  
while(i<len(p_f)):  
# for i in range(0,len(p_f)):  
    if(p_f[i][0]=="MACRO"):  
        funcs.append(p_f[i+1][0])  
        fope=open(p_f[i+1][0],'w+')  
        mac=[]  
        mn=[]  
        al=[]  
        ind_al=1
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
mn.append(mname_index)
mname_index=mname_index+1
mn.append(p_f[i+1][0])
mn.append(index)
al.append(p_f[i+1][0])
if(len(p_f[i+1][1])>0):
    temp_al=p_f[i+1][1].split(",")
    for a in temp_al:
        al.append(str(ind_al)+str(a))
        ind_al=ind_al+1
while(p_f[i][0]!="MEND"):
    temp=[]
    temp.append(index)
    index=index+1
    i=i+1
    for j in p_f[i]:
        temp.append(j)
    mac.append(temp)
    i=i+1
macros.append(mac)
mname.append(mn)
for m in range(1,len(al)):
    fope.write(str(al[m][0]))
    fope.write(" ")
    fope.write(str(al[m][1:]))
    fope.write("\n")
fope.close()
else:
    for m in p_f[i]:
        mac_pass1.write(m)
        mac_pass1.write(" ")
    mac_pass1.write("\n")
    i=i+1
for j in mname:
    for k in j:
        mnt.write(str(k))
        mnt.write(" ")
    mnt.write("\n")
for j in macros:
    for k in j:
        for l in k:
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
        mdt.write(str(l))
        mdt.write(" ")
        mdt.write("\n")
mac_pass1.close()
mdt.close()
mnt.close()
mac_pass=open('macro_pass1.txt','r')
mdt_f=open('mdt.txt','r')
m_f=[]
md_f=[]
for i in mac_pass:
    m_f.append(i.split())
for i in mdt_f:
    md_f.append(i.split())
for i in m_f:
    if(i[0] not in funcs):
        for j in i:
            mac_pass2.write(str(j))
            mac_pass2.write(" ")
        mac_pass2.write("\n")
    else:
        mc_2=[]
        k=0
        args=i[1].split()
        dic={}
        while(k<len(md_f)):
            if(md_f[k][1]==i[0]):
                for t,m in zip(md_f[k][2].split(","),i[1].split(",")):
                    dic[t]=m
                while(md_f[k][1]!="MEND"):
                    if(md_f[k+1][1]!="MEND"):
                        mc_2.append(md_f[k+1])
                        k=k+1
                else:
                    k=k+1
        for k in mc_2:
            for l in range(1,len(k)):
                for m in k[l].split():
                    for n in m.split(","):
                        if(n in dic.keys()):
                            mac_pass2.write(str(dic[n]))
```




K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
        else:
            mac_pass2.write(str(n))
            mac_pass2.write(" ")
        mac_pass2.write("\n")
    func=open(i[0], 'r')
    file_ope=open(i[0]+"F", 'w+')
    f_f=[]
    for h in func:
        f_f.append(h.split())
    for h in f_f:
        for o in h:
            if(o in dic.keys()):
                file_ope.write(str(dic[o]))
            else:
                file_ope.write(str(o))
            file_ope.write(" ")
        file_ope.write("\n")
    file_ope.close()
mac_pass2.close()
print("-----")
print("-----PASS 1-----")
print("-----")
print("MACRO DEFINITION TABLE: ")
table = PrettyTable(['INDEX', 'DEFINITION'])
temp=open('mdt.txt', 'r')
t_f=[]
for i in temp:
    t_f.append(i.split())
t_f2=[]
for i in t_f:
    timepass=[]
    if(len(i)>2):
        timepass.append(i[0])
        strin=""
        for j in range(1, len(i)):
            strin=strin+" "+str(i[j])
        timepass.append(strin)
    else:
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
        for j in i:
            timepass.append(j)
        t_f2.append(timepass)
    for i in t_f2:
        table.add_row(i)
    print(table)
    temp.close()
    print("-----")
    print("MACRO NAME TABLE: ")
    table = PrettyTable(['INDEX', 'MACRO NAME', 'MDT INDEX'])
    temp=open('mnt.txt','r')
    t_f=[]
    for i in temp:
        t_f.append(i.split())
    for i in t_f:
        table.add_row(i)
    print(table)
    temp.close()
    print("-----")
    for i in funcs:
        print("ARGUMENT LIST ARRAY of",i," : ")
        table = PrettyTable(['INDEX', 'ARGUMENT'])
        temp=open(i,'r')
        t_f=[]
        for i in temp:
            t_f.append(i.split())
        for i in t_f:
            table.add_row(i)
        print(table)
        temp.close()
        print("-----")
    print("OUTPUT GENERATED BY PASS1: ")
    temp=open('macro_pass1.txt','r')
    t_f=[]
    for i in temp:
        t_f.append(i.split())
    for i in t_f:
        for j in i:
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
        print(j,end=" ")
    print()
temp.close()
print("-----")
print("-----PASS 2-----")
print("-----")
for i in funcs:
    print("ARGUMENT LIST ARRAY of",i," ")
    table = PrettyTable(['INDEX', 'ARGUMENT'])
    temp=open(i+"F",'r')
    t_f=[]
    for i in temp:
        t_f.append(i.split())
    for i in t_f:
        table.add_row(i)
    print(table)
    temp.close()
    print("-----")
print("OUTPUT GENERATED BY PASS2: ")
temp=open('macro_pass2.txt','r')
t_f=[]
for i in temp:
    t_f.append(i.split())
for i in t_f:
    for j in i:
        print(j,end=" ")
    print()
temp.close()
print("-----")
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

Input files:

Macro_program.txt:

```
macro_program.txt
1  MACRO
2  INCR1 &FIRST,&SECOND
3  A 1,&FIRST
4  L 2,&SECOND
5  MEND
6  MACRO
7  INCR2 &ARG1,&ARG2
8  L 3,&ARG1
9  ST 4,&ARG2
10 MEND
11 PRG2 START
12 USING *,BASE
13 INCR1 DATA1,DATA12
14 INCR2 DATA3,DATA4
15 FOUR DC F'4'
16 FIVE DC F'5'
17 BASE EQU 8
18 TEMP DS '1'F
19 DROP 8
20 END
```

Output:

Files generated:

Mdt.txt:

```
mdt.txt
1 1 INCR1 &FIRST,&SECOND
2 2 A 1,&FIRST
3 3 L 2,&SECOND
4 4 MEND
5 5 INCR2 &ARG1,&ARG2
6 6 L 3,&ARG1
7 7 ST 4,&ARG2
8 8 MEND
9
```



K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Mnt.txt:

```
≡ mnt.txt
1 1 INCR1 1
2 2 INCR2 5
3
```

Macro_pass1.txt:

```
≡ macro_pass1.txt
1 PRG2 START
2 USING *,BASE
3 INCR1 DATA1,DATA12
4 INCR2 DATA3,DATA4
5 FOUR DC F'4'
6 FIVE DC F'5'
7 BASE EQU 8
8 TEMP DS '1'F
9 DROP 8
10 END
11
```

INCR1.txt: (ALA for macros)

```
≡ INCR1
1 1 &FIRST
2 2 &SECOND
3
```

INCR2.txt: (ALA for macros)

```
≡ INCR2
1 1 &ARG1
2 2 &ARG2
3
```



K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Macro_pass2.txt:

```

≡ macro_pass2.txt
1  PRG2 START
2  USING *,BASE
3  A 1 DATA1
4  L 2 DATA12
5  L 3 DATA3
6  ST 4 DATA4
7  FOUR DC F'4'
8  FIVE DC F'5'
9  BASE EQU 8
10 TEMP DS '1'F
11 DROP 8
12 END
13

```

INCR1F.txt: (ALA for macros)

```

≡ INCR1F
1  1 DATA1
2  2 DATA12
3

```

INCR2F.txt: (ALA for macros)

```

≡ INCR2F
1  1 DATA3
2  2 DATA4
3

```

```

-----2 PASS MACROS PROCESSOR-----
-----PASS 1-----
MACRO DEFINITION TABLE:
+-----+
| INDEX | DEFINITION |
+-----+
| 1 | INCR1 &FIRST,&SECOND |
| 2 | A 1,&FIRST |
| 3 | L 2,&SECOND |
| 4 | MEND |
| 5 | INCR2 &ARG1,&ARG2 |
| 6 | L 3,&ARG1 |
| 7 | ST 4,&ARG2 |
| 8 | MEND |
+-----+

```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

MACRO NAME TABLE:

INDEX	MACRO NAME	MDT INDEX
1	INCR1	1
2	INCR2	5

ARGUMENT LIST ARRAY of INCR1 :

INDEX	ARGUMENT
1	&FIRST
2	&SECOND

ARGUMENT LIST ARRAY of INCR2 :

INDEX	ARGUMENT
1	&ARG1
2	&ARG2

OUTPUT GENERATED BY PASS1:

```
PRG2 START
USING *,BASE
INCR1 DATA1,DATA12
INCR2 DATA3,DATA4
FOUR DC F'4'
FIVE DC F'5'
BASE EQU 8
TEMP DS '1'F
DROP 8
END
```

PASS 2

ARGUMENT LIST ARRAY of INCR1 :

INDEX	ARGUMENT
1	DATA1
2	DATA12

ARGUMENT LIST ARRAY of INCR2 :

INDEX	ARGUMENT
1	DATA3
2	DATA4

OUTPUT GENERATED BY PASS2:

```
PRG2 START
USING *,BASE
A 1 DATA1
L 2 DATA12
L 3 DATA3
ST 4 DATA4
FOUR DC F'4'
FIVE DC F'5'
BASE EQU 8
TEMP DS '1'F
DROP 8
END
```

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Conclusion: Understood the concept of macro processors. Learned how macros are expanded in assembly language program also learned how a 2 pass macro processor works. Also implemented the 2 pass macro processor in python.

Post Lab Subjective Questions

1. What are the tasks performed by macro processor?

ANS) Macro represents a group of commonly used statements in the source programming language. Macro Processor replaces each macro instruction with the corresponding group of source language statements. This is known as the expansion of macros. Using Macro instructions programmer can leave the mechanical details to be handled by the macro processor. Macro Processor designs are not directly related to the computer architecture on which it runs. Macro Processor involves definition, invocation, and expansion.

2. What is the purpose of pass1 macro processor?

ANS) The pass 1 of macro processor is used to find the definitions of macros defined in the program. Also pass 1 prepares a intermediate program which will be given as input to the pass2. It also produces macro name table and macro definition table.

3. What is the purpose of pass2 macro processor?

ANS) The pass 2 of macro processor is used to calculate the final program which will be given to the assembler to find the machine code. This pass will recognise the macro calls and replace it with their definitions it will also replaces the arguments which are present in definitions with actual arguments passed when calling.

4. What is the purpose of MDT table?

ANS) This table is a very use full table this table is used to store the definitions of macros defined in the program. This table is referred in both the passes. In pass 1 the main operation to this table is write and in pass 2 it is mainly used for reading only.

5. What is the purpose of MNT table?

ANS) This table used to store the names of the macros with their mdt index. This table will be referred in pass2 to replace call with the actual definition.

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

6. What is the use of MDI & MDLC?

ANS) Additional data structures like macro definition indicator (MDI) and macro definition level counter (MDLC) are the switches used to keep track of macro calls and macro definition. The MDI has status “ON” during the expansion of macro call and the value “OFF” all the other times. When its value is “ON” the cards are read from the MDT and when it is “OFF” the cards are read from the input source card. The use of MDLC is used keep track of the level of macros while defining the macros. Initially it is zero and it is incremented each time a MACRO code is found within a macro. The reverse process happens in case of MEND i.e. the value of MDLC is decremented by one each time it encounters a MEND and the process continues till the MDLC is zero i.e. the completion of macro definition.

Post Lab objective Questions

1. In which way a macro processor for assembly language can be implemented?
 - a. Independent two pass processor
 - b. Independent One pass Processor
 - c. Processor incorporated into pass I of 2-pass Assembler
 - d. All of these above

ANS: d. All of these above

2. Advantage of incorporating the Macro Processor in Pass I of assembler is that
 - a. Many functions have to be implemented twice
 - b. Functions are combined and it is not necessary to create intermediate file as output from the macro processor and input to the assembler.
 - c. More Flexibility is provided to the programmer in that he may use all the features of the assembler in conjunction with macros.
 - d. All of these above.

ANS: d. All of these above.

Date: 3 / 12 / 2021

Signature of faculty in-charge