Rsuita Haldasmalani
(14-210858

EXPERIMENT NO.8

Aim: Write a program to implement Multipass Macroprocessor

Theory:

A multipass macroprocessor is a tool used in computer programming to process source code containing macros. Unlike single pass macroprocessor, which processes macros in a single pass through the source code, a multipass macroprocessor make multiple passes through the resource code to handle more complex macro expansions & dependencies.

1] First Pass: In the first pass, the macroprocessor scans the source code to identify & extract macro definition. Any non-macro code is passed through unchanged.

2] Second pass: On subsequent passes, the macroprocessor goes through the source code again replacing macro invocation with their expansions. They are replaced.

3] Subsequent passes: Depending on the complexity of macros & the need for further processing additional passes may be performed until all macros are expanded & no changes.

Example:

With two macros:

```
; MACRO definitions
DEFINE ADDTWO(X)    MOV 2, X
DEFINE SQUARE(X)    MUL X, X
; Main code
```

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

Start:
ADD_TWO(R1) ; macro invocation
SQUARE (R2) ; macro invocation
HALT ; end of program

First pass:
Identify macro definition & built a symbol table.
- ADD_TWO → MOV Z, X
- SQUARE → MUL X, X

Second pass:
Replace macro invocation with their expansions.
- ADD_TWO (R1) → MOV 2, R1
- SQUARE (R2) → MUL R2, R2

Rewriting code
Start:
MOV 2, R1 ; Expansion of ADD_TWO(R1)
MUL R2, R2 ; Expansion of SQUARE (R2)
HALT ; End of program

Multipass macroprocessor can handle more complex scenarios, such as recursive macro & conditional macro by performing additional passes as needed.
They are used in assembly languages, preprocessors for higher level languages & other contexts where macro are employed.

A+ 7/8/24