INPUT:-

```
MACRO
INCR &ARG1
ADD AREG,&ARG1
MEND
START 100
INCR NUM
NUM DC 5
END
```

Program :-

```
import java.util.*;

public class Ass2 {

    static Scanner sc = new Scanner(System.in);

    // Data Structures for Pass I
    static Map<String, String> mnt = new LinkedHashMap<>(); // Macro Name Table
    static Map<String, List<String>> mdt = new LinkedHashMap<>(); // Macro
Definition Table
    static List<String> ala = new ArrayList<>(); // Argument List Array

    // Pass I: Define Macros
    static void pass1(List<String> inputLines) {
        boolean inMacro = false;
        String macroName = "";
        List<String> macroBody = new ArrayList<>();

        for (String line : inputLines) {
            String[] parts = line.trim().split("\\s+");

            if (parts[0].equalsIgnoreCase("MACRO")) {
                inMacro = true;
                macroBody.clear();
                continue;
            }

            if (inMacro) {
                if (macroName.equals("")) {
                    // First line after MACRO → macro name and parameters
                    macroName = parts[0];
                    if (parts.length > 1) {
                        for (int i = 1; i < parts.length; i++) {
                            ala.add(parts[i]);
                        }
                    }
                    mnt.put(macroName, "MDT#" + (mdt.size() + 1));
                } else if (parts[0].equalsIgnoreCase("MEND")) {
                    // End of Macro
                    inMacro = false;
                    mdt.put(macroName, new ArrayList<>(macroBody));
                    macroName = "";
                } else {
                    // Add macro body line
                    macroBody.add(line);
                }
            }
        }
    }

    // Pass II: Expand macros
```

```
    static void pass2(List<String> inputLines) {
        System.out.println("\n--- Expanded Code (Pass II) ---");
        for (String line : inputLines) {
            String[] parts = line.trim().split("\\s+");
            if (mnt.containsKey(parts[0])) {
                // Macro invocation
                String macroName = parts[0];
                List<String> macroBody = mdt.get(macroName);

                // Replace arguments with actual parameters
                Map<String, String> argMap = new HashMap<>();
                for (int i = 1; i < parts.length; i++) {
                    argMap.put(ala.get(i - 1), parts[i]);
                }

                for (String mline : macroBody) {
                    String expanded = mline;
                    for (String arg : argMap.keySet()) {
                        expanded = expanded.replace(arg, argMap.get(arg));
                    }
                    System.out.println(expanded);
                }
            } else if (!line.equalsIgnoreCase("MACRO") && !
line.equalsIgnoreCase("MEND")) {
                // Normal instruction
                System.out.println(line);
            }
        }
    }

    public static void main(String[] args) {
        // Example input program
        List<String> inputProgram = Arrays.asList(
            "MACRO",
            "INCR &A &B",
            "LOAD &A",
            "ADD &B",
            "STORE &A",
            "MEND",
            "START 100",
            "INCR X Y",
            "END"
        );

        System.out.println("--- Input Program ---");
        for (String line : inputProgram) {
            System.out.println(line);
        }

        // Pass I
        pass1(inputProgram);

        // Display MNT
        System.out.println("\n--- Macro Name Table (MNT) ---");
        for (String key : mnt.keySet()) {
            System.out.println(key + " -> " + mnt.get(key));
        }

        // Display MDT
        System.out.println("\n--- Macro Definition Table (MDT) ---");
        for (String key : mdt.keySet()) {
            System.out.println(key + " : " + mdt.get(key));
        }
```

```
        // Display ALA
        System.out.println("\n--- Argument List Array (ALA) ---");
        for (int i = 0; i < ala.size(); i++) {
            System.out.println(i + " -> " + ala.get(i));
        }

        // Pass II
        pass2(inputProgram);
    }
}
```

OUTPUT:-

```
swaraj@swaraj-VirtualBox:~/LP-1$ javac Ass2.java
swaraj@swaraj-VirtualBox:~/LP-1$ java Ass2
--- Input Program ---
MACRO
INCR &A &B
LOAD &A
ADD &B
STORE &A
MEND
START 100
INCR X Y
END

--- Macro Name Table (MNT) ---
INCR -> MDT#1

--- Macro Definition Table (MDT) ---
INCR : [LOAD &A, ADD &B, STORE &A]

--- Argument List Array (ALA) ---
0 -> &A
1 -> &B

--- Expanded Code (Pass II) ---
LOAD &A
ADD &B
STORE &A
LOAD &A
ADD &B
STORE &A
START 100
LOAD X
ADD Y
STORE X
END
```