

**Roll No. and Name:** 21BCE021 Vani Balani

**Course Code and Name:** 2CS701 Compiler Construction

### **Practical No: 9**

**AIM :**To implement Assembly code generator

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int tempCount = 0;

char* newTemp() {
    static char tempVar[5];
    sprintf(tempVar, "t%d", tempCount++);
    return tempVar;
}

void generateAssembly(char* statement) {
    char lhs[10], op[3], rhs[10], op1[10], op2[10];

    if (strstr(statement, "+=") || strstr(statement, "-=") ||
        strstr(statement, "*=") || strstr(statement, "/=")) {
        sscanf(statement, "%s %s %s", lhs, op, rhs);
        char operation = op[0];

        printf("\n; Assembly for %s %s %s\n", lhs, op, rhs);
        printf("MOV AX, [%s]      ; Load %s into AX\n", lhs, lhs);
        printf("%s AX, [%s]      ; Perform AX %c %s\n",
            (operation == '+') ? "ADD" :
            (operation == '-') ? "SUB" :
            (operation == '*') ? "IMUL" :
            (operation == '/') ? "IDIV" : "NOP", rhs, operation,
            rhs);
        printf("MOV [%s], AX      ; Store result back into %s\n", lhs,
            lhs);
    }
    else if (sscanf(statement, "%s = %s %s %s", lhs, op1, op, op2) ==
        4) {
        printf("\n; Assembly for %s = %s %s %s\n", lhs, op1, op,
            op2);
        printf("MOV AX, [%s]      ; Load %s into AX\n", op1, op1);
```

```

        printf("%s AX, [%s]      ; Perform AX %s %s\n",
               (strcmp(op, "+") == 0) ? "ADD" :
               (strcmp(op, "-") == 0) ? "SUB" :
               (strcmp(op, "*") == 0) ? "IMUL" :
               (strcmp(op, "/") == 0) ? "IDIV" : "NOP", op2, op,
op2);
        printf("MOV [%s], AX      ; Store result back into %s\n", lhs,
lhs);
    }
    else if (sscanf(statement, "%s = %s", lhs, rhs) == 2) {
        printf("\n; Assembly for %s = %s\n", lhs, rhs);
        printf("MOV AX, [%s]      ; Load %s into AX\n", rhs, rhs);
        printf("MOV [%s], AX      ; Store AX into %s\n", lhs, lhs);
    }
    else {
        printf("Invalid statement format.\n");
    }
}


int main() {
    char statement[30];

    printf("Enter an assignment statement");
    fgets(statement, sizeof(statement), stdin);
    statement[strcspn(statement, "\n")] = 0;
    printf("\nGenerated Assembly Code:\n");
    generateAssembly(statement);

    return 0;
}

```

## OUTPUT



```
Enter an assignment statement a = b + c

Generated Assembly Code:

; Assembly for a = b + c
MOV AX, [b]      ; Load b into AX
ADD AX, [c]      ; Perform AX + c
MOV [a], AX      ; Store result back into a

...Program finished with exit code 0
Press ENTER to exit console.
```