**Experiment No. 9 : Simulation of Intermediate Code Generation**

#include <stdio.h>

#include <string.h>

#include <ctype.h>

#define MAX 100

int tempVarCount = 1;

char stack[MAX];

int top = -1;

void push(char c) {

stack[++top] = c;

}

char pop() {

return stack[top--];

}

int precedence(char op) {

if (op == '\*' || op == '/') return 2;

if (op == '+' || op == '-') return 1;

return 0;

}

void generateTAC(char operator, char operand1, char operand2) {

printf("t%d = %c %c %c\n", tempVarCount, operand1, operator, operand2);

push('t' + tempVarCount - 1);

tempVarCount++;

}

void convertToTAC(char \*expr) {

char operands[MAX];

int opTop = -1, i, length = strlen(expr);

char op1, op2, operator;

for (i = 0; i < length; i++) {

if (isalnum(expr[i])) {

push(expr[i]);

} else {

while (top >= 1 && precedence(stack[top - 1]) >= precedence(expr[i])) {

op2 = pop();

operator = pop();

op1 = pop();

generateTAC(operator, op1, op2);

}

push(expr[i]);

}

}

while (top >= 1) {

op2 = pop();

operator = pop();

op1 = pop();

generateTAC(operator, op1, op2);

}

}

int main() {

char expr[MAX];

printf("Enter an expression: ");

scanf("%s", expr);

convertToTAC(expr);

return 0;

}

**Output:**



