9 - 3 Address Code

Tuesday, 19 March 2024 10:07 AM

Ashima Fatima Seik Mujibur Rahman, 21BAI1830

AIM: Write a program to generate 3- address code for the given expression.

Sample Input

x = a + (b * c) - d

Sample Output

t1 = b * c
t2 = t1 - d
t3 = a + t2
x = t3

Procedure

1. Input the expression in the required format
2. Parse the input expression to extract the variables and operators by tokenizing the input based on spaces. Then store them in seperate variables
3. Initialize temoporary variables to store the intermediate results.
4. Check the operator variables to check operator priority to generate the 3- address code accordingly.
5. print the 3- address code.

Sample Input

x = a + b * c - d

Code Output

```
Enter the expression: t1=b*c
t2=a+t1
x =t2
```

Code :

```c
#include <stdio.h>
#include <string.h>
// Function to generate 3-address code
void generate3AddressCode(char *expression) {
    char t1[10], t2[10], t3[10];
    char *x, *var1, *var2, *var3, *op1, *op2;
    // Parsing the expression
    x = strtok(expression, "=");
```

```
    var1 = strtok(NULL, " ");
    op1 = strtok(NULL, " ");
    var2 = strtok(NULL, " ");
    op2 = strtok(NULL, " ");
    var3 = strtok(NULL, " ");
    // Generating 3-address code based on operator priorities
    if (strcmp(op2, "*") == 0 || strcmp(op2, "/") == 0) {
        printf("t1=%s%s%s\n", var2, op2, var3);
        printf("t2=%s%st1\n", var1, op1);
        printf("%s=t2\n", x);
    }
    else {
        printf("t1=%s%s%s\n", var1, op1, var2);
        printf("t2=%s%st1\n", var3, op2);
        printf("%s=t2\n", x);
    }
}
int main() {
    char expression[100];
    // Input
    printf("Enter the expression: ");
    fgets(expression, sizeof(expression), stdin);
    // Remove newline character from the input
    expression[strcspn(expression, "\n")] = 0;
    // Generate 3-address code
    generate3AddressCode(expression);
    return 0;
}
```