Om Shete
THADOMAL SHAHANI TSEC ENGINEERING COLLEGE
TE C32
2103163

Experiment No : 5

Aim : WAP to implement Three Address Code

Theory :

- Three address code (TAC) is an intermediate type of code which is easy to generate and can be easily converted to machine code.
- It makes use of at most three addresses and one operator to represent an expression and the value computed at each instruction is stored in temporary variable generated by compiler.
- The compiler decides the order of operation given by three address code.
- Three address code is used in compiler applications :-

1) **Optimization**
⇒ TAC is often used as an intermediate representation of code during optimization phases of compilation process.

2) **Code generation**
⇒ TAC code can also be used as an intermediate representation of code during the code generation phase of the compilation process.

3) **Debugging**
⇒ TAC can be helpful in debugging the code

1) Quadraple Field

⇒

operator
source 1
source 2
Destination

$p = -q * r + s$

$t1 = -q$

$t2 = r + s$

$t3 = t1 * t2$

$p = t3$

2) Triple Field

⇒

Operator
Source 1
Source 2

$p = -q * r + s$

$t1 = -q$

$t2 = r + s$

$t3 = t1 * t2$

$p = t3$

generated by the compiler. Since three address code is low-level language, it is often easier to read and understand the final generated code.

4) Language translation

⇒ TAC can also be used to translate code from one programming language to another.

- Three representation technique :-
  → Quadruple
  → Triples
  → Indirect tripler

- General Illustration :

  a = b op c

  a, b, c → operands

  op → operator

- Conclusion : In this program I learned about the TAC and how to implement it.

28/03/2024

Code:

```
import re


print("enter your choice 1 for assignment 2 for arithmetic 3 for
relational 4 to exit")
choice = int(input())
while choice != 4:
    if choice == 1:
        # assignment
        print("enter the variable")
        var = input()


        print(var.split("="))
        print("t1 = ", var.split("=")[1])
        print(var.split("=")[0], "= t1")
    elif choice == 2:
        print("enter the expression")
        exp = input()
        i = 0
        while i < len(exp):
            if exp[i] == "+" or exp[i] == "-":
                if exp[i + 2] == "*" or exp[i + 2] == "/":
                    print("t1 = ", exp[i + 1], exp[i + 2], exp[i + 3])
                    print("t2 = ", exp[i - 1], exp[i], "t1")
                    break
                else:
                    print("t1 = ", exp[i - 1], exp[i], exp[i + 1])
                    print("t2 = t1", exp[i + 2], exp[i + 3])
                    break
            elif exp[i] == "*" or exp[i] == "/":
                if exp[i + 2] == "+" or exp[i + 2] == "-":
                    print("t1 = ", exp[i - 1], exp[i], exp[i + 1])
                    print("t2 = t1", exp[i + 2], exp[i + 3])
                    break
            elif exp[i] == "*" or exp[i] == "/":
                print("t1 = ", exp[i - 1], exp[i], exp[i + 1])
                break
            i += 1
    elif choice == 3:
        print("enter the relational expression")
```

```
        exp = input().split(" ")
        operators = ["<", ">", "<=", ">=", "==", "!="]
        if exp[1] in operators:
            print("100 IF ", exp[0], exp[1], exp[2], " GOTO 103")
            print("101 T:=0")
            print("102 GOTO 104")
            print("103 T:=1")
            print("104")
    else:
        print("invalid operator")
    print(
        "enter your choice 1 for assignment 2 for arithmetic 3 for
relational 4 to exit"
    )
    choice = int(input())
```

Output:

```
Run                                          Ask AI   41s on 00:46:36, 03/27
enter your choice 1 for assignment 2 for arithmetic 3 for relational 4 to exit
1
enter the variable
a=d
['a', 'd']
t1 =  d
a = t1
enter your choice 1 for assignment 2 for arithmetic 3 for relational 4 to exit
2
enter the expression
a=b+c*d
t1 =  c * d
t2 =  b + t1
enter your choice 1 for assignment 2 for arithmetic 3 for relational 4 to exit
3
enter the relational expression
a > b
100 IF  a > b  GOTO 103
101 T:=0
102 GOTO 104
103 T:=1
104
enter your choice 1 for assignment 2 for arithmetic 3 for relational 4 to exit
4
```