Date: 30/10/2023

Roll No. and Name: Yash Ginoya (20BCE075)

Course Code and Name: Compiler Construction (2CS701)

Practical No.: 6

# Aim:

Intermediate Code Generation: To generate Three Address code for assignment statement

---------------------------------------------------------------------------------------------------

**Input Files:**

* **Practical6.cpp**

#include <bits/stdc++.h>

using namespace std;

int prec(char c)

{

    if (c == '^')

        return 3;

    else if (c == '/' || c == '\*')

        return 2;

    else if (c == '+' || c == '-')

        return 1;

    else

        return -1;

}

string infixToPostfix(string s)

{

    stack<char> st;

    string result;

    for (int i = 0; i < s.length(); i++) {

        char c = s[i];

        if ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z')

            || (c >= '0' && c <= '9'))

            result += c;

        else if (c == '(')

            st.push('(');

        else if (c == ')') {

            while (st.top() != '(') {

                result += st.top();

                st.pop();

            }

            st.pop();

        }

        else {

            while (!st.empty()

                && prec(s[i]) <= prec(st.top())) {

                result += st.top();

                st.pop();

            }

            st.push(c);

        }

    }

    while (!st.empty()) {

        result += st.top();

        st.pop();

    }

    return result;

}

int main()

{

//  string s = "x=a+b\*(c^d-e)^(f+g\*h)-i";

    // string s = "x=y\*(z+a)";

    string s;

    cout<<"enter statement : ";

    cin>>s;

    string var = "";

    int i = 0;

    while(i<s.size()){

        if(s[i] == '=') break;

        else var.push\_back(s[i++]);

    }

    s.erase(s.begin(),s.begin()+i+1);

    string res = infixToPostfix(s);

//  cout<<res<<endl<<endl;

    stack<string> st;

    int ind = 0;

    vector<string> ans;

    int cnt = 1;

    while(ind<res.size()){

        if(res[ind] == '+' || res[ind] == '-' || res[ind] == '\*' || res[ind] == '/' || res[ind] == '^'){

            string t1 = st.top();

            st.pop();

            string t2 = st.top();

            st.pop();

            string temp = "t" + to\_string(cnt) + "=" + t2 + res[ind] + t1;

            ans.push\_back(temp);

            st.push("t" + to\_string(cnt));

            cnt++;

        }

        else{

            string zuzu = "";

            zuzu.push\_back(res[ind]);

            st.push(zuzu);

        }

        ind++;

    }

    string str = var;

    str += "=t" + to\_string(--cnt);

    ans.push\_back(str);

    for(auto it:ans){

        cout<<it<<endl;

    }

    return 0;

}

**Output:**

****

****

**Conclusion:**

After performing this practical, I learn how to generate Intermediate Code Generation To generate Three Address code for assignment statement.

---------------------------------------------------------------------------------------