Practical No 6

Aim: Write a code to generate the DAG for the input arithmetic expression.

```java
import java.util.ArrayList;
import java.util.List;

public class DAG {

    private String expression = "B*-C+B*-C";
    private List<String> tacList = new ArrayList<String>();
    String[][] m = new String[10][10];

    public void printExpression() {
        System.out.println("Input Expression :");
        System.out.println("===================");
        System.out.println("Expression = " + expression );
    }

    public void convertToTAC() {
        int cnt = 1;
        String tempStr = new String();
        tempStr = "T" + cnt + "=" + expression.substring(2, 4);
        m[0][0] = "T" + cnt;
        m[0][1] = null;
        m[0][2] = expression.substring(2, 3);
        m[0][3] = expression.substring(3, 4);

        cnt++;

        tacList.add(tempStr);
        tempStr = "T" + cnt + "=" + expression.substring(0, 2) + "T" + (cnt - 1);
        m[1][0] = "T" + cnt;
        m[1][1] = expression.substring(0, 1);
        m[1][2] = expression.substring(1, 2);
        m[1][3] = "T" + (cnt - 1);

        cnt++;

        tacList.add(tempStr);
        tempStr = "T" + cnt + "=" + expression.substring(7, 9);
        m[2][0] = "T" + cnt;
        m[2][1] = null;
        m[2][2] = expression.substring(7, 8);
        m[2][3] = expression.substring(8, 9);

        cnt++;

        tacList.add(tempStr);
        tempStr = "T" + cnt + "=" + expression.substring(5, 7) + "T" + (cnt - 1);
        m[3][0] = "T" + cnt;
```

```java
            m[3][1] = expression.substring(5, 6);
            m[3][2] = expression.substring(6, 7);
            m[3][3] = "T" + (cnt - 1);

            cnt++;

            tacList.add(tempStr);
            tempStr = "T5=T2+T4";
            m[4][0] = tempStr.substring(0, 2);
            m[4][1] = tempStr.substring(3, 5);
            m[4][2] = tempStr.substring(5, 6);
            m[4][3] = tempStr.substring(6, 8);
            tacList.add(tempStr);

            System.out.println("Three Address Code : ");
            System.out.println("======================");
            for (int i = 0; i < tacList.size(); i++) {
                System.out.println("  " + tacList.get(i));
            }
            System.out.println();
            System.out.println(" Temporary variable count : " + tacList.size() + "\n\n");
            System.out.println(" Temporary variables :");
            System.out.println(" ====================");
            for (int i = 0; i < tacList.size(); i++) {
                System.out.println("  T" + (i + 1));
            }
            System.out.println("\n");
            System.out.println("Label  |  Left Child  |   Operator  |  Right Child  ");
            System.out.println("=====================================================");
            for (int i = 0; i < 5; i++) {
                for (int j = 0; j < 4; j++) {
                    System.out.print(m[i][j] + "\t \t");
                }
                System.out.println();
            }

    }

    public static void main(String[] args) {
        DAG dg = new DAG ();
        System.out.println("Practical Performed by Rasika Sawant");
        dg.printExpression();
        dg.convertToTAC();

    }
}
```

```
Output - DAG (run)   ×

run:
Practical Performed by Rasika Sawant
Input Expression :
====================
Expression = B*-C+B*-C
Three Address Code :
======================
  T1=-C
  T2=B*T1
  T3=-C
  T4=B*T3
  T5=T2+T4


 Temporary variable count : 5



 Temporary variables :
 ======================
  T1
  T2
  T3
  T4
  T5



 Label  |  Left Child  |   Operator  |  Right Child
 ============================================================
 T1            null          –               C
 T2            B             *               T1
 T3            null          –               C
 T4            B             *               T3
 T5            T2            +               T4
 BUILD SUCCESSFUL (total time: 0 seconds)
```