

# System Programming and Operating Systems Lab

## ASSIGNMENT 10

**Name: Shrirang Mhalgi.**

**Roll No: 322008**

**Batch: B1**

### **1 Date of Completion:**

20/03/2019

### **2 Aim:**

Implement Dynamic Link Library for mathematical operation and test the same.

### **3 Objectives:**

To study Dynamic Link Library implementation in Java.

### **4 Theory:**

Dynamic-link library (or DLL) is Microsoft's implementation of the shared library concept in the Microsoft Windows and OS/2 operating systems. These libraries usually have the file extension DLL, OCX (for libraries containing ActiveX controls), or DRV (for legacy system drivers). The file formats for DLLs are the same as for Windows EXE files that is, Portable Executable (PE) for 32-bit and 64-bit Windows, and New Executable (NE) for 16-bit Windows. As with EXEs, DLLs can contain code, data, and resources, in any combination.

A DLL file is often given a ".dll" file name suffix. DLL files are dynamically linked with the program that uses them during program execution rather than being compiled into the main program.

The advantage of DLL files is space is saved in random access memory (RAM) because the files don't get loaded into RAM together with the main program. When a DLL file is needed, it is loaded and run. For example, as long as a user is editing a document in Microsoft Word, the printer DLL file does not need to be loaded into RAM. If the user decides to print the document, the Word application causes the printer DLL file to be loaded and run.

A program is separated into modules when using a DLL. With modularized components, a program can be sold by module, have faster load times and be updated without altering other parts of the program. DLLs help operating systems and programs run faster, use memory efficiently and take up less disk space.

## 5 Code:

### Java Program

```
import java.util.Scanner;
public class CalcJNI
{
    static
    {
        System.loadLibrary(" Calc ");
    }

    private native void doCalculate (int a,int b,char ch);

    public static void main(String [] args)
    {
        int a,b;
        char ch;
        Scanner sc = new Scanner ( System.in ); System.
        out.println(" Enter          first number:" );
        a=sc.nextInt();
        System.out.println(" Enter second number:" );
        b=sc.nextInt();
        System.out.println(" Enter Operation('+' '-' '*' '/' :)" );
        ch=sc.next().charAt(0);
        new CalcJNI ().doCalculate ( a,b,ch );
    }
}
```

## Header File:

```
/* DO NOT EDIT THIS FILE - it is machine generated */
#include <jni.h>
/* Header for class CalcJNI */

#ifdef Included_Calc_JNI
#define Included_Calc_JNI
#ifdef _cplusplus
extern "C" {
#endif
/*
 * Class:      CalcJNI
 * Method:     doCalculate
 * Signature:  (IIC)V
 */
JNIEXPORT void JNICALL Java_Calc_JNI_doCalculate
    (JNIEnv *, jobject, jint, jint, jchar);

#ifdef _cplusplus
}
#endif
#endif
```

## C Program:

```
#include <jni.h>
#include <stdio.h>
#include "CalcJNI.h"

JNIEXPORT void JNICALL Java_Calc_JNI_doCalculate
(JNIEnv *env, jobject thisObj, jint a, jint b, jchar ch)
{
    int n;
    printf(" First Number:%d",a);
    printf("\nSecond Number:%d",b);
    switch (ch)
    {
        case '+' :      n=a+b ;
                        printf("\nAddition:%d\n",n);
                        break ;
        case '-' :      n=a-b ;
                        printf("\nSubtraction:%d\n",n);
                        break ;
        case '*' :      n=a*b ;
                        printf("\nMultiplication:%d\n",n);
                        break ;
        case '/' :      n=a/b ;
                        printf("\nDivision:%d\n",n);
                        break ;
        default : printf("\nError occurred!\n");
    }
    return ;
}
```

## 6 Output:

```
comp-7@comp7:~/Documents/Sarthak$ javac CalcJNI.java
comp-7@comp7:~/Documents/Sarthak$ javah CalcJNI
comp-7@comp7:~/Documents/Sarthak$ gcc -I/usr/lib/jvm/java-6-openjdk-amd64/include -o libCalc.so -shared -fPIC CalcJNI.c
comp-7@comp7:~/Documents/Sarthak$ java -Djava.library.path=. CalcJNI
Enter first number:
5
Enter second number:
4
Enter Operation('+ '-' '*' '/'):
*
First Number:5
Second Number:4
Multiplication:20
comp-7@comp7:~/Documents/Sarthak$ |
```

## 7 Conclusion:

In this assignment we learn in detail the concept and implementation of Dynamic Link Library.