

## 임베디드시스템실습\_lab11

2017313107 이승태

### 1. JNI를 이용한 간단한 사칙연산 함수 만들기

Main.java에서 실행되어야할 함수들과 System.loadLibrary로 Cal을 불러와줬다.

```
1 public class Main
2 {
3     public static void main(String[] args)
4     {
5         System.loadLibrary("Cal");
6         new Cal().sum(1,2);
7         new Cal().difference(1,2);
8         new Cal().product(1,2);
9         new Cal().average(1,2);
10    }
11 }
```

Cal.java에서는 함수의 형태만 정의해준다.

```
1 public class Cal
2 {
3     public native int sum(int x, int y);
4     public native int difference(int x, int y);
5     public native int product(int x, int y);
6     public native double average(int x, int y);
7 }
```

실제 코드는 C에서 구현하고 Cal.h에 나와있는 함수 형태로 그대로 구현해준다.

```
1 #include <stdio.h>
2 #include "Cal.h"
3
4 JNIEXPORT jint JNICALL Java_Cal_sum(JNIEnv *env, jobject obj, jint a, jint b)
5 {
6     printf("%d\n", a + b);
7     return a + b;
8 }
9
10 JNIEXPORT jint JNICALL Java_Cal_difference(JNIEnv *env, jobject obj, jint a, jint b)
11 {
12     printf("%d\n", a - b);
13     return a - b;
14 }
15
16 JNIEXPORT jint JNICALL Java_Cal_product(JNIEnv *env, jobject obj, jint a, jint b)
17 {
18     printf("%d\n", a * b);
19     return a * b;
20 }
21
22 JNIEXPORT jdouble JNICALL Java_Cal_average(JNIEnv *env, jobject obj, jint a, jint b)
23 {
24     printf("%f\n", (double)(a + b)/2);
25     return (double)((a + b)/2);
26 }
```

Main을 실행하면 다음과 같이 사칙연산이 실행된다.

int a = 1

int b = 2

```
패키지 openjdk-8-jdk-headless은 이미 최신 버전입니다 (80292-b10-0000101~20.04).
0개 업그레이드, 0개 새로 설치, 0개 제거 및 103개 업그레이드 안 함.
lee@lee-GF75-Thin-9SC:~/java$ JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
lee@lee-GF75-Thin-9SC:~/java$ gcc -shared -fPIC -I $JAVA_HOME/include -I $JAVA_H
OME/include/linux -o libCal.so Cal.c
lee@lee-GF75-Thin-9SC:~/java$ export LD_LIBRARY_PATH=.
lee@lee-GF75-Thin-9SC:~/java$ java Main
3
-1
2
1.500000
lee@lee-GF75-Thin-9SC:~/java$
```

2. 버튼을 눌렀을 때, segment값을 editText에 있는 숫자로 바꾸어 주는 코드이다.  
native-lib에서 syscall을 정의해 주었다.

```
1  #include <jni.h>
2  #include <string>
3  #include <unistd.h>
4
5  extern "C" JNIEXPORT void JNICALL
6  Java_com_example_myapplication_MainActivity_segmentset(JNIEnv *env, jobject thiz, jint num) {
7      syscall( number: 323, num);
8      return;
9  }
```

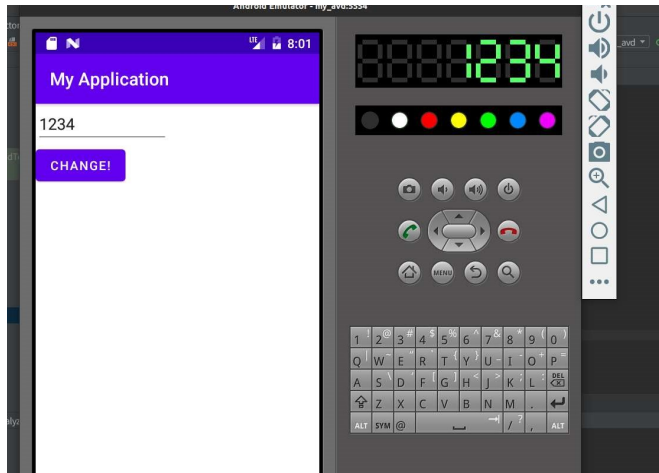
MainActivity에서 onClick을 정의해줘서 button이 클릭 되었을 때, c코드로 짜인 segmentset이 불린다.

```
1  public class MainActivity extends AppCompatActivity {
2
3      // Used to load the 'native-lib' library on application startup.
4      static {
5          System.loadLibrary( libname: "native-lib");
6      }
7
8      private EditText tv;
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_main);
14         tv = findViewById(R.id.text);
15     }
16
17     @
18     public void onClick(View view) {
19         if (view.getId() == R.id.button) {
20             segmentset(Integer.parseInt(tv.getText().toString()));
21         }
22     }
23
24     /**
25      * A native method that is implemented by the 'native-lib' native library,
26      * which is packaged with this application.
27      */
28     public native void segmentset(int num);
29 }
```

xml코드이다. EditText와 Button을 정의해줬다.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".MainActivity"
8      android:orientation="vertical">
9
10     <EditText
11         android:id="@+id/text"
12         android:layout_width="150dp"
13         android:layout_height="wrap_content"
14         android:inputType="numberDecimal"/>
15
16     <Button
17         android:id="@+id/button"
18         android:layout_width="wrap_content"
19         android:layout_height="wrap_content"
20         android:text="Change!"
21         android:onClick="onClick"
22     />
23
24 </LinearLayout>
```

1234를 넣고 CHANGE!버튼을 누르면 다음과 같이 1234로 바뀐다.



Edit text에 1234351값을 넣고 CHANGE! 버튼을 누르면 1234351로 바뀐다.

