ECE 150/251 Assignment 3

Name: Shenghan Gao

App description:

There are only 1 activity in this app. A SurfaceView, 3 SeekBars and a Switch are used in the user interface. In this app, the frequency of the wave can be set from 0 to 40Hz, the smoothing factor can be set from 0 to 0.1.

When the app is launched, it is in the sine wave mode, the 2 signals are drawn on the SurfaceView, but you can only find 1 line because now the 2 signals overlap. After both the signal amplitude bar and frequency bar are moved so that they are no longer 0, the original red signal will be shown on the SurfaceView and it will be seen to be changing continuously if the amplitude or frequency set by the SeekBars is changed. Now if the smoothing factor is changed to a value between 0 and 0.1, the filtered green signal can also be seen. Basically, if the frequency of the wave increases, the amplitude of the filtered signal will decrease. Also, the filtered wave will change once the smoothing factor bar is dragged.

If the Switch is turned on, the app will be changed into cosine wave mode and the behavior of the app is similar to that of the sine wave mode.

Meanwhile, some debug information will be printed to logcat, that is, once the sine wave generating function is called, two lines of debug information ("Sine wave function is called." and "Another way: Sine wave function is called.") will be shown in logcat using different ways.

Answers to the questions:

1. In order to print out messages to logcat, we need to include <android/log.h> and use the function there, also, we need to modify build.gradle so that there will be a line stating the log library which is automatically generated in Android.mk. The code is like the following:

Code example (Important parts are in gold):

main.cpp

#include <jni.h>

#include <android/log.h>

#ifndef LOG\_TAG

#define LOG\_TAG "JNIDebug"

#define LOGD(...) \_\_android\_log\_print(ANDROID\_LOG\_DEBUG, LOG\_TAG, \_\_VA\_ARGS\_\_)

#endif

#include <cmath>

#include "com\_shenghangao\_filterit\_MainActivity.h"

extern "C" {

JNIEXPORT jdoubleArray JNICALL Java\_com\_shenghangao\_filterit\_MainActivity\_sineWave (JNIEnv \*env, jobject jObj, jdouble width, jdouble height,jdouble amplitude, jdouble frequency)

{

LOGD("Sine wave function is called.");

\_\_android\_log\_print(ANDROID\_LOG\_DEBUG, "JNIDebug", "Another way: Sine wave function is called.");

jdoubleArray result = env->NewDoubleArray(1005);

jdouble y[1005];

jdouble interval = 0.001;

for (int i=0; i<1005; ++i)

{

y[i] = amplitude/100 \* height/2 \* sin(2\*M\_PI\*frequency\*i\*interval);

}

env->SetDoubleArrayRegion(result, 0, 1004, y);

return result;

}

JNIEXPORT jdoubleArray JNICALL Java\_com\_shenghangao\_filterit\_MainActivity\_cosineWave (JNIEnv \*env, jobject jObj, jdouble width, jdouble height,jdouble amplitude, jdouble frequency)

{

jdoubleArray result = env->NewDoubleArray(1005);

jdouble y[1005];

jdouble interval = 0.001;

for (int i=0; i<1005; ++i)

{

y[i] = amplitude/100 \* height/2 \* cos(2\*M\_PI\*frequency\*i\*interval);

}

env->SetDoubleArrayRegion(result, 0, 1004, y);

return result;

}

JNIEXPORT jdoubleArray JNICALL Java\_com\_shenghangao\_filterit\_MainActivity\_LPF (JNIEnv \*env, jobject jObj, jdoubleArray lpfin, jdouble factor)

{

int len = env->GetArrayLength(lpfin);

jdoubleArray result = env->NewDoubleArray(len);

jdouble y[len];

jdouble \*input = env->GetDoubleArrayElements(lpfin, 0);

y[0] = input[0];

for (int i=1; i<len; ++i)

{

y[i] = factor \* input[i] + (1-factor) \* y[i-1];

}

env->SetDoubleArrayRegion(result, 0, len-1, y);

env->ReleaseDoubleArrayElements(lpfin, input, 0);

return result;

}

}

build.gradle

apply plugin: 'com.android.application'

android {

compileSdkVersion 21

buildToolsVersion "21.1.2"

defaultConfig {

applicationId "com.shenghangao.filterit"

minSdkVersion 19

targetSdkVersion 21

versionCode 1

versionName "1.0"

ndk {

moduleName "Jnilib"

ldLibs "log"

}

}

buildTypes {

release {

minifyEnabled false

proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'

}

}

}

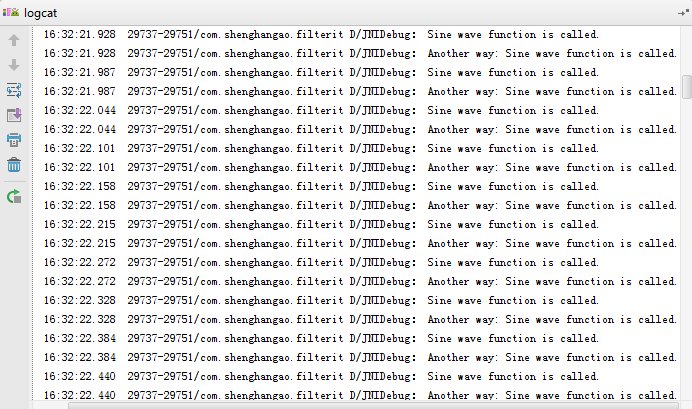
dependencies {

compile fileTree(dir: 'libs', include: ['\*.jar'])

compile 'com.android.support:appcompat-v7:21.0.3'

}

The result in logcat after I making use of LOG\_TAG to filter the messages:



The picture shows that the program works.