

JNI Laboraufgabe 1a

Android-Studio-Projekt "LaborJNI"

Erstelle in der vorgegebenen Activity eine Methode, die den Text einer TextView aktualisiert. Z.B. mit *"Hello from C++"*. Die Activity enthält schon eine Java-Methode dafür:

```
public void setWelcomeText(String newText)
```

Deklariere in der Klasse `MainActivity` eine native Methode, die in `onCreate` aufgerufen werden soll, und implementiere sie per JNI:

```
public native void welcomeFromJNI();
```

JNI-Anleitung: lokalisiere in der Projektansicht unter "app" den Ordner "cpp" mit der Datei "native-lib.cpp". Dort ist bereits ein Platz mit der Signatur der nativen Methode vorgesehen.

Benutze folgende `JNIEnv`-Methoden, um von C++ aus in Java o.g. Methode `setWelcomeText(String newText)` aufzurufen:

```
jclass JNIEnv::GetObjectClass(jobject obj)
jmethodID JNIEnv::GetMethodID(jclass clazz, const char* name,
const char* sig)
jstring JNIEnv::NewStringUTF(const char* bytes)
void CallVoidMethod(jobject obj, jmethodID methodID, ...)
```

Der `GetMethodID`-Signatur-Parameter `char* sig` benötigt folgendes Format:

"(argument-types) return-type"

Signature	Java Type
V	void (nur ret-type)
Z	boolean
B	byte
C	char
S	short
I	int
J	long
F	float
D	double
L fully-qualified-class ;	fully-qualified-class
[type	type[]
(arg-types) ret-type	method type

Beispiel 1: "(I)F" für die Signatur: `float ToInt(int i)`

Beispiel 2: "(Ljava/lang/String;)I" für die Signatur: `String ParseInt(int i)`

JNI Laboraufgabe 1b

Android-Studio-Projekt "LaborJNI"

JNI Laboraufgabe 1b (optional für Enthusiasten):

Der initial auf der TextView angezeigte Text soll per JNI **erweitert** werden (string concatenation). Dafür sind in der Activity zwei Java-Methoden vorgesehen:

```
public String getWelcomeText()  
public void setWelcomeText(String newText) // wie in Aufgabe 1a
```

Implementiere eine zweite native Methode und rufe sie in `onCreate` auf.

```
public native void appendWelcomeFromJNI();
```

Anleitung:

Der Aufbau des C-Funktionsnamens zunächst von `welcomeFromJNI` abgucken.

Für die Implementierung muss vor dem Aufruf von `setWelcomeText(String newText)` der bestehende Text mit `getWelcomeText()` abgefragt werden. Die Verbindung von zwei Strings kann in C++ z.B. so realisiert werden:

```
// concatenate two std::string  
std::string newStdString = stdString1 + stdString2;
```