# JNI

* JNI java本地开发接口
* JNI 是一个协议
* 这个协议用来沟通java代码和外部的本地代码(c/c++).
* 通过这个协议,java代码就可以调用外部的c/c++代码
* 外部的c/c++代码也可以调用java代码
* JNI怎么用：
  + 1、JAVA语言
  + 2、C/C++语言
  + 3、java jni流程
  + 4、NDK (native develop kits )
* 使用Java调用jni，jni再去调用c写的模块
* 音频播放的java api中，就是使用的jni去调用c实现的

# C语言：

## 开发环境搭建

* 安装DEV-C++ 5，这是轻量级的。
* 执行的是c语言1999的标准
* Android中也是1999的标准

## Helloword

## C代码调用java代

* Java需要编译为class文件

# C的基本数据类型

* char, int, float, double, long, short, signed, unsigned
* int:
  + 32位，能表示的数字是2的32次方个
  + 最高位用来表示符号位，那么还剩下31位可以表示数值，所以能表示的数字就是2的31次方个
  + int的表示范围：
    - 当最高位是0时：0~2147483647
    - 当最高位是1是：-2147483648~-1
* C语言中char不能表示中文，因为它只有一个字节，中文有4个字节
* C语言中，使用char数组表示字符串
  + char arr[] = “adfsa”
  + char arr[] = {‘a’, ’b’, ’c’}
  + char arr[4]
  + c中，数组越界也没关系，不会检查越界的。照样能存
  + char\* cstr = “afasdfsf”;，这是jni中 常用的定义字符串的方法
* signed:有符号位
* unsigned：无符号为
* java的数据类型长度：字节
  + byte：1
  + short：2
  + int：4
  + long：8
  + boolean：1
  + char：2
  + float：4
  + double：8
* c的基本数据类型长度
  + short：2
  + int：4
  + long：4
  + char：1
  + float：4
  + double：8
  + long long : 8，整型
  + short int：2，整型，没实际意义
  + long int：4，整型，美实际意义
* c的char在java中用byte表示，因为他们长度一样
* c的long在java中用int表示
* java的boolean在c中用char表示
* java的long在c中用long long表示

# C的输出函数

* %d - int，整型
* %ld – long int，长整型
* %hd – 短整型
* %c - char
* %f - float，自动补齐六位小数，如果超过六位，四舍五入变成六位小数
* %lf – double，自动补齐六位小数，如果超过六位，四舍五入变成六位小数
* %u – 无符号数，最高为表示符号（正负）的位，变成表示数字
* %x – 十六进制输出 int 或者long int 或者short int
* %#x – 十六进制输出，补充十六进制格式：0x22323
* %o - 八进制输出
* %#o – 八进制输出，补充八进制的格式 025245
* %s – 字符串

# C的输入函数

# 指针

* 指针就是内存地址
* Int I = 3;
* 会开辟一个内存空间
* 这个内存空间里面有3
* 这个内存空间有个内存地址（指针）
* I是内存空间的别名
* 每个字节都有一个内存地址
* 可以通过软件，锁定内存地址，更改里面的值。或者锁定他的值不变（无限血的原理）

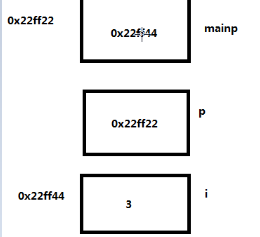
# 指针变量

* 指针就是地址，地址就是指针
* 地址就是内存单元的编号
* 指针变量是存放地址的变量
* 指针和指针变量是两个不同的概念
* 但是要注意： 通常我们叙述时会把指针变量简称为指针，实际它们含义并不一样
* Int\* p; 没有赋值的指针变量，是野指针
* 不能为野指针所表示的那个地址中的内容赋值：\*p = 23; 这是不行的
* 因为，野指针指向的地址不知道是哪里，是系统随机分配的。如果赋值，会导致数据错乱

# 值传递和返回多个值

# 主函数中获取子函数变量的内存地址

看视频



# 数组入门

# 指针的长度

* int a = 13;
* int\* i = &a;
* printf(“%d\n”, sizeof(i));
* 所有指针变量的长度都是4个字节
* 在数组中，内存地址是连续的，所以指针可以相加减
* 如果内存地址不连续，不能相加减

# 堆栈概念：

* 栈：
* 由系统自动分配.例如,声明一个局部变量int  b; 系统自动在栈中为b开辟空间.例如当在调用涵数时，需要保存的变量，最明显的是在递归调用时，要系统自动分配一个栈的空间，后进先出的，而后又由系统释放这个空间.
* 堆：
  + 需要程序员自己申请，并指明大小，在c中用malloc函数
  + 如char\* p1  =  (char\*) malloc(10);   //14byte， p1四个字节 + 右边10个字节
  + 但是注意p1本身是在栈中的.
  + Java中，new的就是在堆中
* 栈上分配的内存，编译器会自动收回；堆上分配的内存，要通过free来显式地收回,否则会造成内存泄漏。
* Java中，堆也会被垃圾回收机制自己回收
* 堆和栈的区别可以用如下的比喻来看出：
* 使用栈就像我们去饭馆里吃饭，只管点菜（发出申请）、付钱、和吃（使用），吃饱了就走，不必理会切菜、洗菜等准备工作和洗碗、刷锅等扫尾工作，他的好处是快捷，但是自由度小。
* 使用堆就像是自己动手做喜欢吃的菜肴，比较麻烦，但是比较符合自己的口味，而且自由度大。
* ppt24页

# 动态内存分配

* 动态内存分配，就是在堆内存中分配内存空间
* 静态就是，在栈内存中，分配内存空间

# 学号管理

# 多级指针

看视频

# 结构体

* 跟对象类似

# 联合体

# 枚举

# 自定义类型

# Jni.h

* C和java之间，数据类型之间做翻译的范本
* 具体看视频