

aaudio 范例: 调用 C 语言之弹性数组

```
//aaudio 调用 C 语言之弹性数组
import tcc;
tcc.build( "/.flexible_array.dll").code = /**
#include <stdlib.h>
#include <string.h>

typedef struct{
    int length;
    char bytes[];
} TestStruct;

__declspec(dllexport) TestStruct* createTestStruct(){
    TestStruct *ts = (TestStruct *) malloc (sizeof (TestStruct) + 100);
    ts->length = strlen("测试一下");
    strcpy(ts->bytes, "测试一下");
    return ts;
}

__declspec(dllexport) void freeTestStruct(TestStruct* p){
    free(p);
}
**/

//加载生成的DLL
var dll = raw.loadDll( "/.flexible_array.dll",,"cdecl" );

//方法1
var pStruct = dll.createTestStructP();

//首先得到弹性数组的长度
var header = raw.convert(pStruct,{int length});

//获取弹性数组
var struct = raw.convert(pStruct,{
    int length;
    BYTE bytes[/*不能指定变量值*/] = {
        length=header.length; //弹性数组的长度必须用 length 属性指定
    }
});

//上面的两步也可以合并为下面的一句代码
var struct = raw.convert(pStruct,{
    int length;
    BYTE bytes[] = raw.convert(pStruct,{int length; /*如果是结构体数组, 这里放一个结构体 —— 作为数组元素类型声明*/})
});

import console;
console.log( string.pack( struct.bytes ) );

//也可以直接计算指针地址, 直接获取数据
var struct = raw.convert(pStruct,{int length});
var offset = raw.sizeof({int length});
var str = raw.toString(pStruct,offset,offset + struct.length);
console.log( str );

dll.freeTestStruct(pStruct);

console.pause(true);
```