aardio 范例: 调用 C 语言 - 结构化参数表

```
//aardio 调用 C 语言 - 结构化参数表

import tcc;
var c = tcc();
c.enableIoPrintf();

c.code = /****
    #include <stdio.h>
    #include <stdlib.h>

    //在C语言中定义 raw.cdeclParameter,注意 aardio 字符串默认为 UTF8 编码
    typedef struct{
        const char *(__cdecl *getType) (const char * name);
        void * (__cdecl *getFunction) (const char * name,const char *proto);
        void (__cdecl *setFunction) (const char * name,const char *proto,void * addr);
        const char *(__cdecl *getBinary) (const char * name,unsigned int *size);
        void (__cdecl *setBinary) (const char * name,char * value,unsigned int size);
        const char *(__cdecl *getString) (const char * name);
        void (__cdecl *setString) (const char * name,const char * value);
        void (__cdecl *getNumber) (const char * name,double * value);
        void (__cdecl *setNumber) (const char * name,double value);
        unsigned long long (__cdecl *getSize64) (const char * name);
        void (__cdecl *setSize64) (const char * name,unsigned long long value);
        void * (__cdecl *getPointer) (const char * name);
        void (__cdecl *setPointer) (const char * name,void * value);
        int (__cdecl *callString) (const char * name,const char * arg);
        int (__cdecl *callNumber) (const char * name,double arg);
        int (__cdecl *call) (const char * name);
        unsigned int(__cdecl *len)(const char * name);
    } aardioParameter;

    typedef double (*ADDFUNC) (double a,double b);

    int func_c ( aardioParameter * opt )
    {

        //取参数中的字段值,字段名可以使用名字空间,例如  x.y.z.字段名
        const char * s = opt->getString("hello");
        io_printf( "Hello! 我是C语言代码\n收到aardio传来的参数:%s\n", s );

        //调用参数中包含的函数名
        opt->callString("func","参数");

        //可以添加C函数为 aardio 函数
        opt->setFunction("test.printf","void(string s,int x,int y)",printf);

        //也以声明aardio中的函数为C函数
        int (*add) (int a,int b) =  opt->getFunction("test.add","int(int,int)" );
        int c = (*add)(12,3);

        unsigned long long  x =  opt->getSize64("size" );
        io_printf( " LONG64: %I64u\n", x);
        return 0;
    }

****/



//创建结构化参数
import console;
import raw.cdeclParameter;
var cdeclParameter = raw.cdeclParameter(
    size = ..math.size64(2,1);
    hello = "测试!";
    func = function(参数){
        ..console.log("aardio函数被回调了",参数 )
    }
    test = {
        add  = function(a,b){
            owner.printf( '在aardio中调用C语言声明的函数 %d %d\n',12,33 );
            return a+b
        }
    }
)
```

```
//获取C函数
func_c = c.getCdecl("func_c","int(struct msg)")

//调用C函数
func_c( cdeclParameter )

//关闭C语言编译器
c.close();

console.pause();
```

Markdown 格式