

《计算机游戏开发》

Cocos2d-x 中文输入问题解决方案

解决方案 1 (By 黄嘉雯):

在 MapScene.h 中添加网上提供的中文封装类:

```
#ifndef _CHINESE_DISPLAY_H_ //自寻的显示中文的封装类
#define _CHINESE_DISPLAY_H_

#include <iostream>
#include "stringapiset.h"
#include "winerror.h"
using namespace std;

class chineseDisplay
{
public:
    static string a(char *b)
    {
        wstring cc = str2wstr(b);

        return WideByte2UTF8(cc);
    }
private:
    static wstring str2wstr(const string &str)
    {
        setlocale(LC_ALL, "chs"); //设当前程序的地域设置为“chs”(中文)，影响全部内容。
        const char* _Source = str.c_str(); //c_str()函数返回一个指向正规C字符串的指针，内容与本string串相同
        size_t _Dsize = str.size() + 1;
        wchar_t *_Dest = new wchar_t[_Dsize]; //wchar_t是字符类型，char为8位，wchar_t一般为16位或32位
        wmemset(_Dest, 0, _Dsize); //将wchar_t的0到wchar_t数组的_Dest[]第一位中
        mbstowcs(_Dest, _Source, _Dsize); //将char数据转到wchar_t类型的_Dest[]里
        wstring result = _Dest; //wstring是宽字符串，占2字节。
        delete[] _Dest; //清空_Dest
        setlocale(LC_ALL, "C"); //恢复为原本的地域设置。

        return result; //返回宽字符类型的中文数据
    }

    static string WideByte2UTF8(const wstring& text)
    {
        int asciisize = ::WideCharToMultiByte(CP_UTF8, 0, text.c_str(), text.size(), NULL, 0, NULL, NULL); //UNICODE型的字串转成ANSI
        if (asciisize == ERROR_NO_UNICODE_TRANSLATION || asciisize == 0) //防出错
        {
            return string();
        }
        char* resultstring = new char[asciisize];
        int convresult = ::WideCharToMultiByte(CP_UTF8, 0, text.c_str(), text.size(), resultstring, asciisize, NULL, NULL); //转换完
        if (convresult != asciisize) //防出错
        {
            return string();
        }
        std::string buffer(resultstring, convresult); //生成为字符串
        delete[] resultstring; //清空resultstring

        return buffer; //输出字符串
    }
};

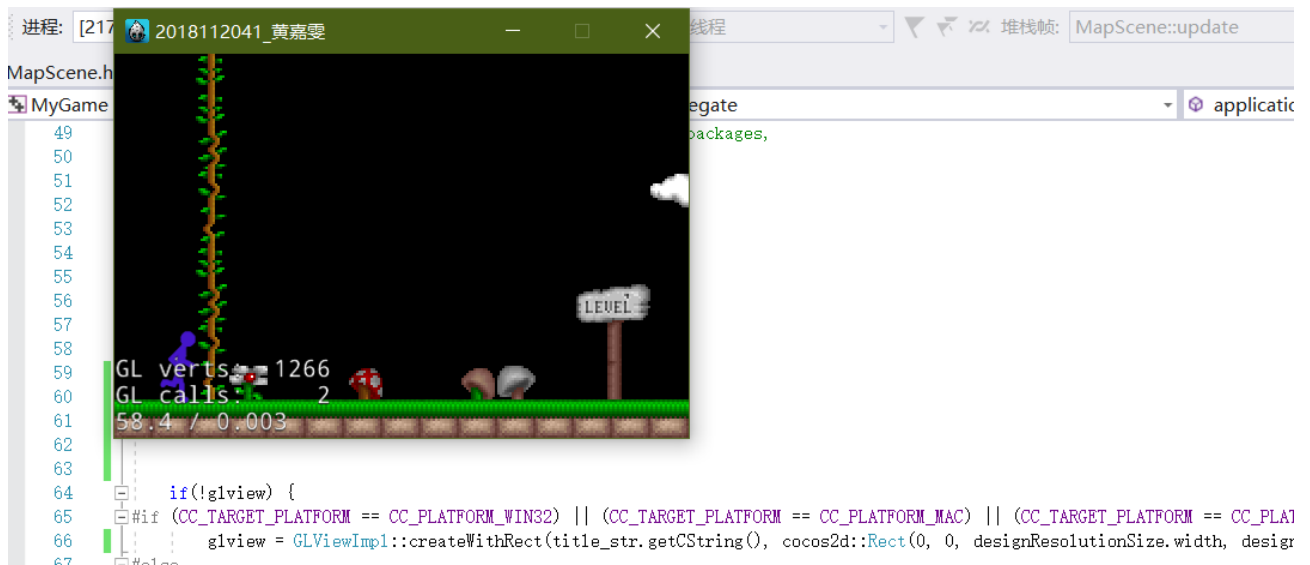
#endif
```

然后在 AppDelegate.cpp 中修改代码，即可调用它实现显示中文，如图：

(新增紫色块，黄色代替原本的字符串"MyGame".)

```
57 // initialize director
58 auto director = Director::getInstance();
59 auto glview = director->getOpenGLView();
60
61 char Title_name[] = "2018112041_黄嘉雯";
62 String title_str = chineseDisplay::a(Title_name);
63
64
65 if(!glview) {
66 #if (CC_TARGET_PLATFORM == CC_PLATFORM_WIN32) || (CC_TARGET_PLATFORM == CC_PLATFORM_MAC) || (CC_TARGET_PLATFORM == CC_PLATFORM_LINUX)
67     glview = GLViewImpl::createWithRect(title_str.getCString(), cocos2d::Rect(0, 0, designResolutionSize.width, designResolutionSize.height));
68 #else
69     glview = GLViewImpl::create(title_str.getCString());
70 #endif
71     director->setOpenGLView(glview);
72 }
```

运行结果如下：



解决方案 2 (By 钟毅):

cocos2d-x 自带的 iconv 库可以让我们轻松的实现中文显示。

运用 iconv 库需要图 8 中函数

```
GBKToUTF8(hello, "gb2312", "utf-8");
```

处理中文字符串，使其可以正常使用。

```
(CC_TARGET_PLATFORM == CC_PLATFORM_WIN32) || (CC_TARGET_PLATFORM == CC_PLATFORM_MAC) || (CC_TARGET_PLATFORM == CC_PLATFORM_LINUX)
std::string hello = "实验一：钟毅，2019285081";
GBKToUTF8(hello, "gb2312", "utf-8");
glview = GLViewImpl::createWithRect(hello, cocos2d::Rect(0, 0, designResolutionSize.width, designResolutionSize.height));
```

图 8



图 9

GBKToUTF8() 函数的定义在新添加的 tools.cpp 和 tools.h 文件里面。两文件的代码如图 10 和图 11 所示，并且因为调用了 iconv 库所以需要在 VS 里面添加 iconv 库的引用。使用的 VS 是 2015 年版本的，添加 iconv 需要进行三步操作，如图 12 图、13、图 14 所示。

图 12：在项目属性中，C/C++ 目录下的常规，添加附加包含目录：D:\cocos2d-x-3.17.2\external\win32-specific\icon\include

图 13：在项目属性中，连接器目录下的常规，添加附加库目录：D:\cocos2d-x-3.17.2\external\win32-specific\icon

图 14：在项目属性中，连接器目录下的输入，添加附加附加依赖项：libiconv.lib
完成以上步骤后，就可以轻松实现 cocos2d-x 中的中文字符串显示。

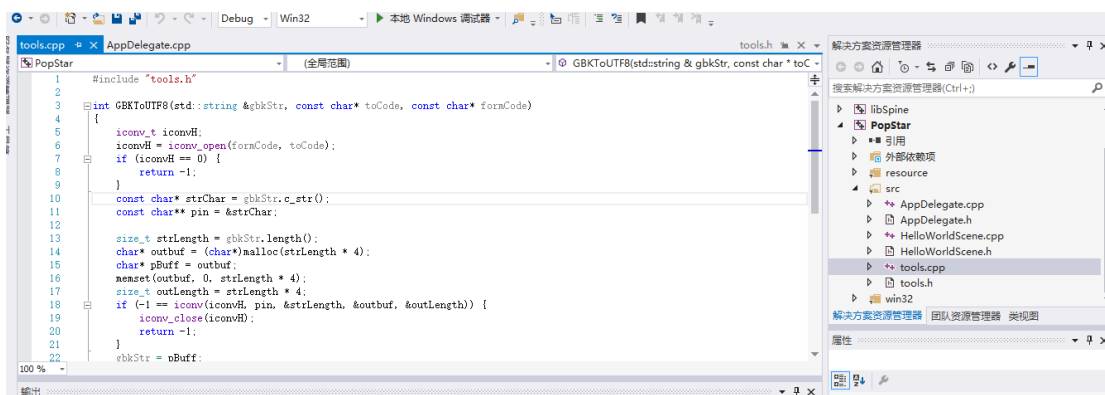


图 10

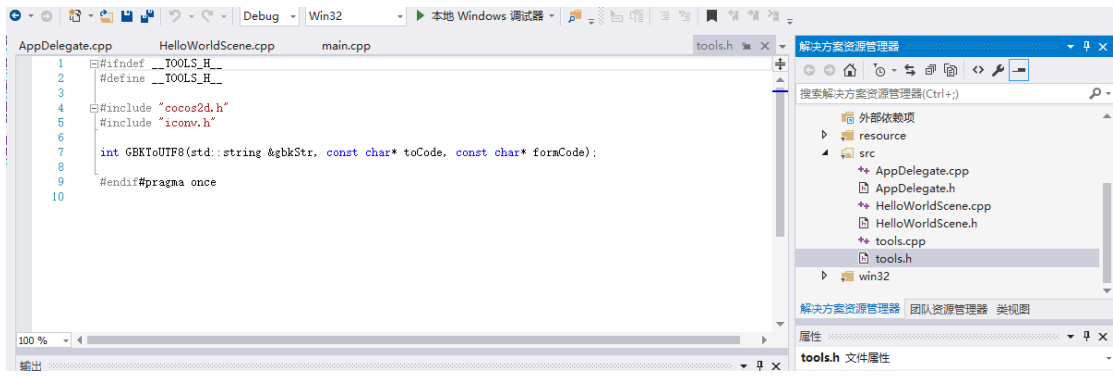


图 11

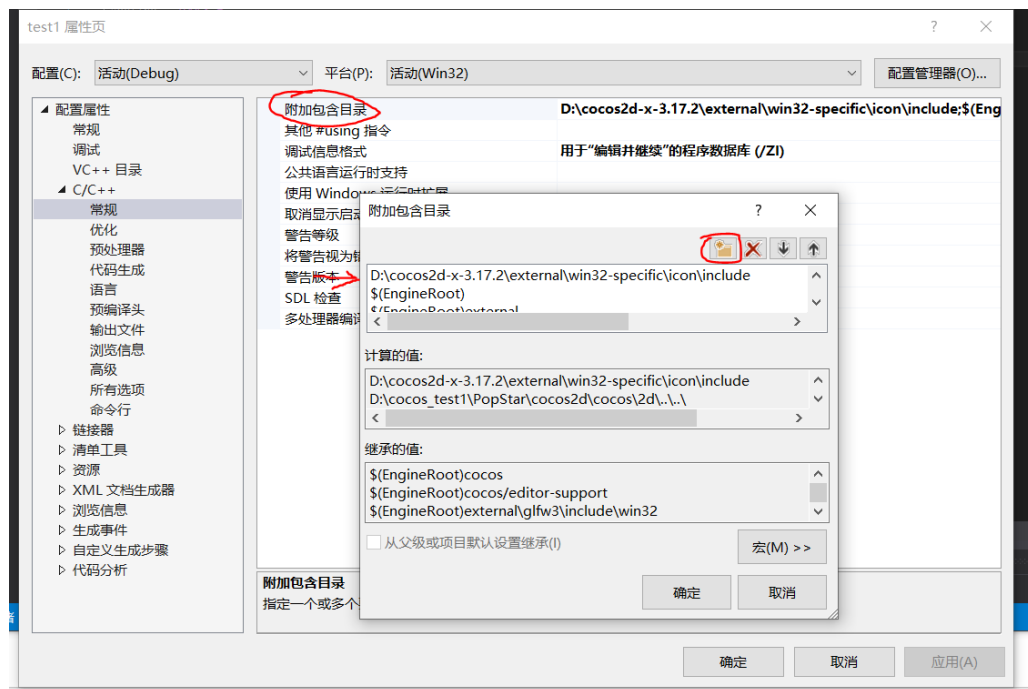
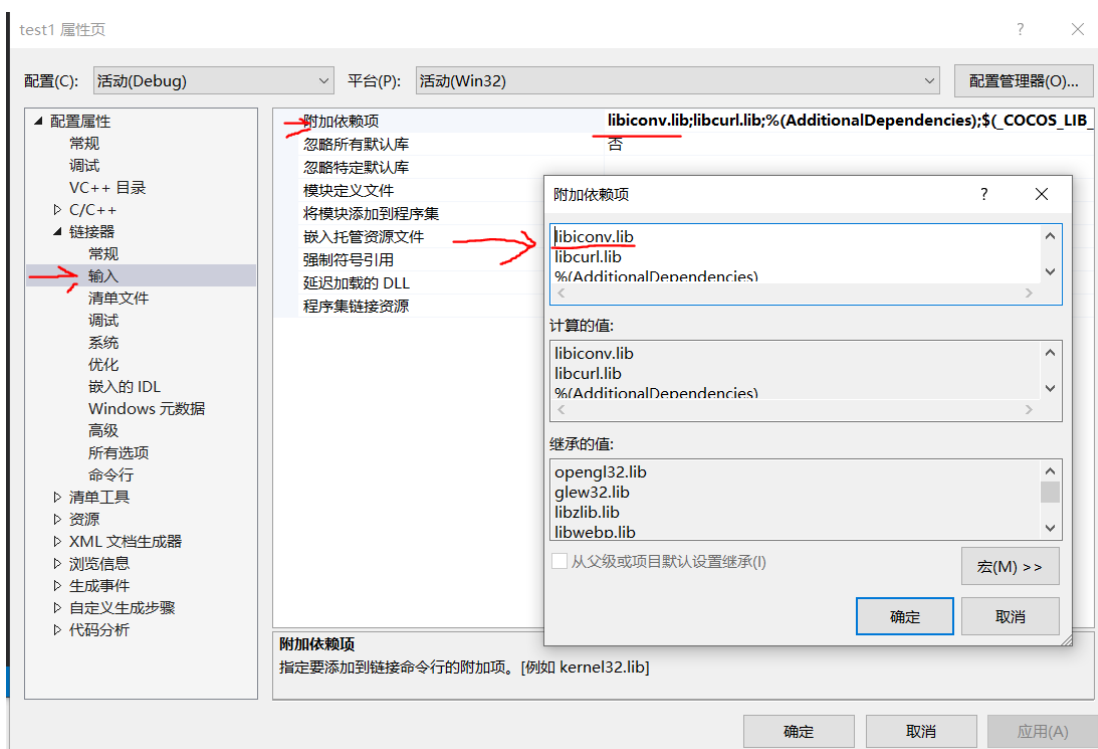
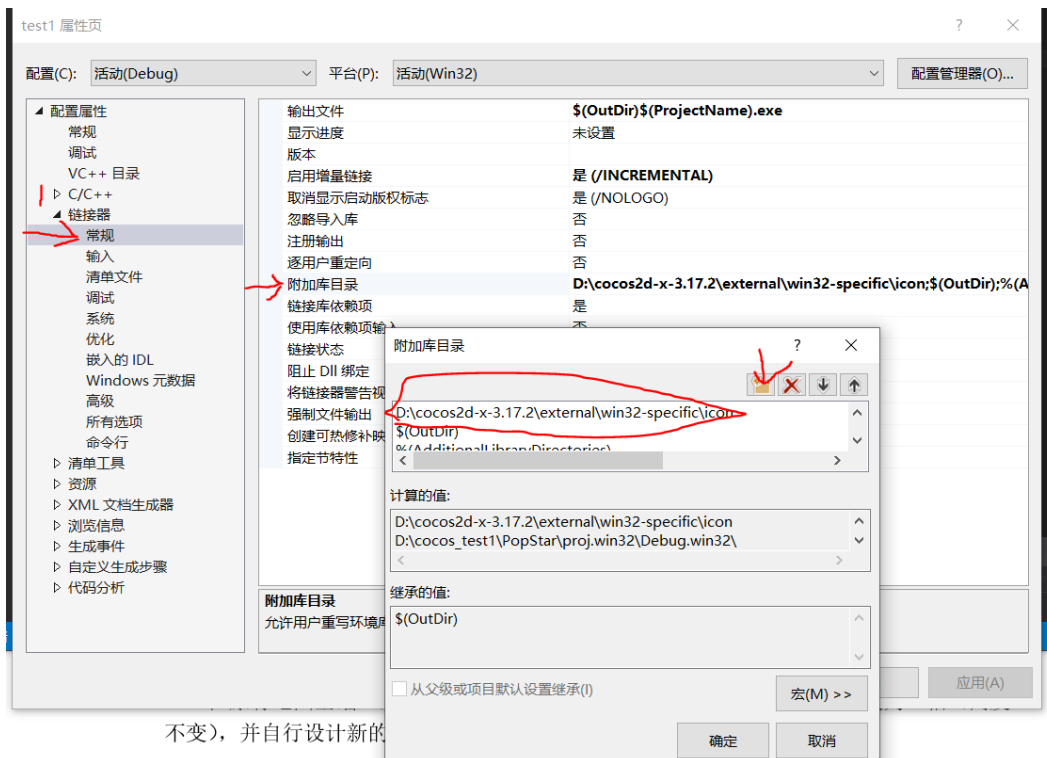


图 12



解决方案 3 (By 林泓昊):

通过查阅资料可知, 由于 VisualStudio 的默认编码方式为 GBK 格式, 而 cocos2d-x 引擎的默认编码格式为 UTF-8 格式, 故中文会出现无法兼容的情况, 所以需要将 GBK 编码转为 UTF-8 编码。

通过查阅资料, 得到 GBK 转 UTF-8 代码:

```
//用于转换中文字符 (GBK转UTF-8)
std::string GBKToUTF8(const std::string &strGBK)
{
    string strOutUTF8 = "";
    WCHAR *str1;
    int n = MultiByteToWideChar(CP_ACP, 0, strGBK.c_str(), -1, NULL, 0);
    str1 = new WCHAR[n];
    MultiByteToWideChar(CP_ACP, 0, strGBK.c_str(), -1, str1, n);
    n = WideCharToMultiByte(CP_UTF8, 0, str1, -1, NULL, 0, NULL, NULL);
    char *str2 = new char[n];
    WideCharToMultiByte(CP_UTF8, 0, str1, -1, str2, n, NULL, NULL);
    strOutUTF8 = str2;
    delete[]str1;
    str1 = NULL;
    delete[]str2;
    str2 = NULL;
    return strOutUTF8;
}
```

原初始化修改为:

```
bool AppDelegate::applicationDidFinishLaunching() {
    // initialize director
    auto director = Director::getInstance();
    auto glview = director->getOpenGLView();
    if(!glview) {
        return false;
    }
    if (CC_TARGET_PLATFORM == CC_PLATFORM_WIN32) || (CC_TARGET_PLATFORM == CC_PLATFORM_MAC) || (CC_TARGET_PLATFORM == CC_PLATFORM_LINUX)
    {
        glview = GLViewImpl::createWithRect("2018192004_林泓昊_HeroRun", cocos2d::Rect(0, 0, designResolutionSize.width, designResolutionSize.height));
    }
    director->setOpenGLView(glview);

    return true;
}
```

编译运行程序:

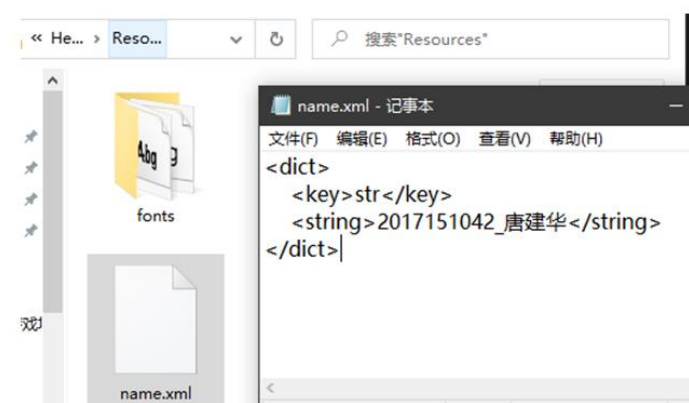


成功显示中文字符。

解决方案 4 (By 唐建华):

修改游戏显示名称

- ① 由于 cocos2dx 使用中文会出现乱码, 所以需要写一个 xml 文件放在 Resources 文件夹内, 如下。



② 接着在 AppDelegate.cpp 中引入 xml，获取其中的文字内容赋值到 str，实现如下。

```
bool AppDelegate::applicationDidFinishLaunching() {
    // initialize director
    auto director = Director::getInstance();
    auto glview = director->getOpenGLView();
    auto *chnStrings = Dictionary::createWithContentsOfFile("name.xml");
    const char *str = ((String*)chnStrings->objectForKey("str"))->getCString();
    if (!glview) {
        if (CC_TARGET_PLATFORM == CC_PLATFORM_WIN32) || (CC_TARGET_PLATFORM == CC_PLATFORM_MAC) || (CC_TARGET_PLATFORM == CC_PLATFORM_LINUX)
            glview = GLViewImpl::createWithRect(str, cocos2d::Rect(0, 0, designResolutionSize.width, designResolutionSize.height));
        else
            glview = GLViewImpl::create(str);
    }
    director->setOpenGLView(glview);
}
```

③ 运行游戏，查看窗口名字，如下。



注意事项：一定将 xml 文件编码格式选为 UTF-8（如下图所示）。

