## 1.产生原因：

|  |
| --- |
|  |

## 2.解决办法，

<1>在需要显示中文的cpp文件的开头加一个预编译指令：#pragma execution\_character\_set("utf-8")

<2>将中文保存到一个xml文件中，然后使用Dictionary来加载：

如，有一个zhcn.xml文件，内容如下

|  |
| --- |
| <dict>  <key>hello</key>  <string>你好</string>  <key>win</key>  <string>你赢了！！！</string>  <key>quit</key>  <string>确定退出吗？</string>  </dict> |

读取代码：

|  |
| --- |
| auto dict = Dictionary::createWithContentsOfFile("zhcn.xml");  auto str = ((String\*)dict->objectForKey("hello"))->getCString();  log(str); |

控制台输出：你好

win10下面plist文件中有中文，cocosed：：FileUtils就解析不出来，

不要使用plist文件和FileUtils，在win10它应该没有用，加载不到有中文的文件

**注意：在vs中所看到的目录有时候是虚拟目录，有叫做筛选器，目的是使用项目结构工具清晰，它其实并不会改变文件的实际路径**

**可以使用xml+Dictionary的方式**

**再来一个实例：**

**有一个中文集文件内容如下：tips.xml,它只有一个根元素dict，其他的都是key和string（value），都是字符串**

|  |
| --- |
| **<?xml version="1.0" encoding="UTF-8"?>**  **<dict>**  **<key>bgkstory</key>**  **<string>游戏故事背景</string>**  **<key>start</key>**  **<string>开始</string>**  **<key>regret</key>**  **<string>悔棋</string>**  **<key>actlost</key>**  **<string>认输</string>**  **<key>exit</key>**  **<string>退出</string>**  **<key>readdoc</key>**  **<string>是否读取存档</string>**  **<key>yes</key>**  **<string>是</string>**  **<key>no</key>**  **<string>否</string>**  **<key>vscom</key>**  **<string>与电脑对战?</string>**  **<key>blorwh</key>**  **<string>选择黑棋还是白棋</string>**  **<key>blkchess</key>**  **<string>黑棋</string>**  **<key>whchess</key>**  **<string>白棋</string>**  **<key>uwon</key>**  **<string>你赢了!</string>**  **<key>ulost</key>**  **<string>你输了!</string>**  **<key>blkwin</key>**  **<string>黑棋胜利!</string>**  **<key>whwin</key>**  **<string>白棋胜利</string>**  **</dict>** |

**遍历输出中文字符串的代码**

|  |
| --- |
| auto dict = Dictionary::createWithContentsOfFile("tips.xml");  auto keys = dict->allKeys();    for (int i = 0; i < keys->count();i++) {  auto obj = keys->getObjectAtIndex(i);  auto k=((String\*)obj)->getCString();  log(((String\*)dict->objectForKey(k))->getCString());  } |

**结果，注意这个字典是没有序的**

|  |
| --- |
| 悔棋  游戏故事背景  与电脑对战?  开始  认输  是否读取存档  退出  是  否  选择黑棋还是白棋  黑棋  白棋  你赢了!  你输了!  黑棋胜利!  白棋胜利 |

## 3.在cocos2d程序在加载xml文件

3.1创建一个场景类，名称叫做ChineseDemo，继承自Secene，需要引入cocos2d头文件和命名空间

|  |  |
| --- | --- |
| //ChineseDemo.h  #pragma once  #include "cocos2d.h"  USING\_NS\_CC;  #include<map>  using namespace std;  class ChineseDemo:public Scene  {  public:    static Scene\* createScene();  virtual bool init();  CREATE\_FUNC(ChineseDemo);  private:  //记录中文字集合key:中文编号value:中文字  map<int, Value> ptr\_cnWords;  }; | //ChineseDemo.cpp  #include "ChineseDemo.h"  //#pragma execution\_character\_set("utf-8") //设置vs2017支持中文  Scene \* ChineseDemo::createScene()  {  return ChineseDemo::create();//create()方法其实是会自动调用init方法，也就是说初始化的功能其实是在init中实现的，只不过暴露给外面使用的是create方法  }  bool ChineseDemo::init()  {  //先调用父类的init方法  if (!Scene::init()) {  //如果父类初始化失败就返回false  return false;  }  auto visibleSize = Director::getInstance()->getVisibleSize();  Vec2 origin = Director::getInstance()->getVisibleOrigin();    /\*auto dict = Dictionary::createWithContentsOfFile("zhcn.xml");  auto str = ((String\*)dict->objectForKey("hello"))->getCString();  log(str);\*/  auto dict = Dictionary::createWithContentsOfFile("tips.xml");  auto keys = dict->allKeys();    //遍历输出  /\*for (int i = 0; i < keys->count();i++) {  auto obj = keys->getObjectAtIndex(i);  auto k=((String\*)obj)->getCString();  log(((String\*)dict->objectForKey(k))->getCString());  }\*/    //我们只需要有自己项目需要的就行  auto str = ((String\*)dict->objectForKey("bgkstory"))->getCString();  auto label = Label::createWithSystemFont(str,"Arial",36);  label->setAnchorPoint(Vec2::ANCHOR\_MIDDLE);  label->setPosition(Vec2(visibleSize.width/2,visibleSize.height /2));  this->addChild(label);  return true;  } |

#pragma execution\_character\_set(“utf-8”)