

cocos2dx

1、环境搭建

1.1、编译环境

- Mac OS X 10.7+, Xcode 5.1+
- Ubuntu 14.04+, gcc 4.9+, CMake 2.6+
- Windows 7+, VS 2013+
- Python 2.7.5
- NDK r11+
- Android SDK API Level 19
- JDK 1.6+

1.2、 android SDK & NDK

- 安装android studio, 下载地址:

<https://dl.google.com/dl/android/studio/install/2.3.3.0/android-studio-ide-162.4069837-mac.dmg>

- 安装完成后, 设置SDK, 以下是SDK勾选项

SDK Platforms SDK Tools SDK Update Sites				
Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.				
	Name	API Level	Revision	Status
<input type="checkbox"/>	Google APIs Intel x86 Atom System Image	21	22	Not installed
<input type="checkbox"/>	Google APIs Intel x86 Atom_64 System Image	21	22	Not installed
▼ <input type="checkbox"/>	Android 4.4W (KitKat Wear)			
<input type="checkbox"/>	Android SDK Platform 20	20	2	Not installed
<input type="checkbox"/>	Sources for Android 20	20	1	Not installed
▼ <input checked="" type="checkbox"/>	Android 4.4 (KitKat)			
<input checked="" type="checkbox"/>	Google APIs	19	20	Installed
<input type="checkbox"/>	Glass Development Kit Preview	19	11	Not installed
<input checked="" type="checkbox"/>	Android SDK Platform 19	19	4	Installed
<input checked="" type="checkbox"/>	Sources for Android 19	19	2	Installed
<input type="checkbox"/>	ARM EABI v7a System Image	19	5	Not installed
<input checked="" type="checkbox"/>	Intel x86 Atom System Image	19	6	Installed
<input type="checkbox"/>	Google APIs ARM EABI v7a System Image	19	30	Not installed
<input checked="" type="checkbox"/>	Google APIs Intel x86 Atom System Image	19	30	Installed

SDK Platforms SDK Tools SDK Update Sites			
Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.			
	Name	Version	Status
▼	Android SDK Build-Tools		
<input type="checkbox"/>	17.0.0 (Obsolete)	17.0.0	Not installed
<input type="checkbox"/>	18.0.1 (Obsolete)	18.0.1	Not installed
<input type="checkbox"/>	18.1.0 (Obsolete)	18.1.0	Not installed
<input type="checkbox"/>	18.1.1 (Obsolete)	18.1.1	Not installed
<input type="checkbox"/>	19.0.0 (Obsolete)	19.0.0	Not installed
<input type="checkbox"/>	19.0.1 (Obsolete)	19.0.1	Not installed
<input type="checkbox"/>	19.0.2 (Obsolete)	19.0.2	Not installed
<input type="checkbox"/>	19.0.3 (Obsolete)	19.0.3	Not installed
<input checked="" type="checkbox"/>	19.1.0	19.1.0	Installed

- 下载解压NDK，下载地址：

https://dl.google.com/android/repository/android-ndk-r14b-darwin-x86_64.zip

- 设置环境变量

```
# Add android SDK
export ANDROID_SDK_ROOT=/yourpath/android-sdk
export PATH=$PATH:$ANDROID_SDK_ROOT/tools
export PATH=$PATH:$ANDROID_SDK_ROOT/platform-tools

# Add android NDK
export NDK_ROOT=/yourpath/android-ndk-r15c
export PATH=$PATH:$NDK_ROOT
```

2、Cocos2d-x

2.1 安装

- 下载源文件并解压

<http://www.cocos2d-x.org/filedown/cocos2d-x-3.15.1.zip>

- 执行setup.py设置环境变量

```
cd /yourpath/cocos2d-x-3.15.1
./setup.py
```

- 执行CLI查看是否安装成功

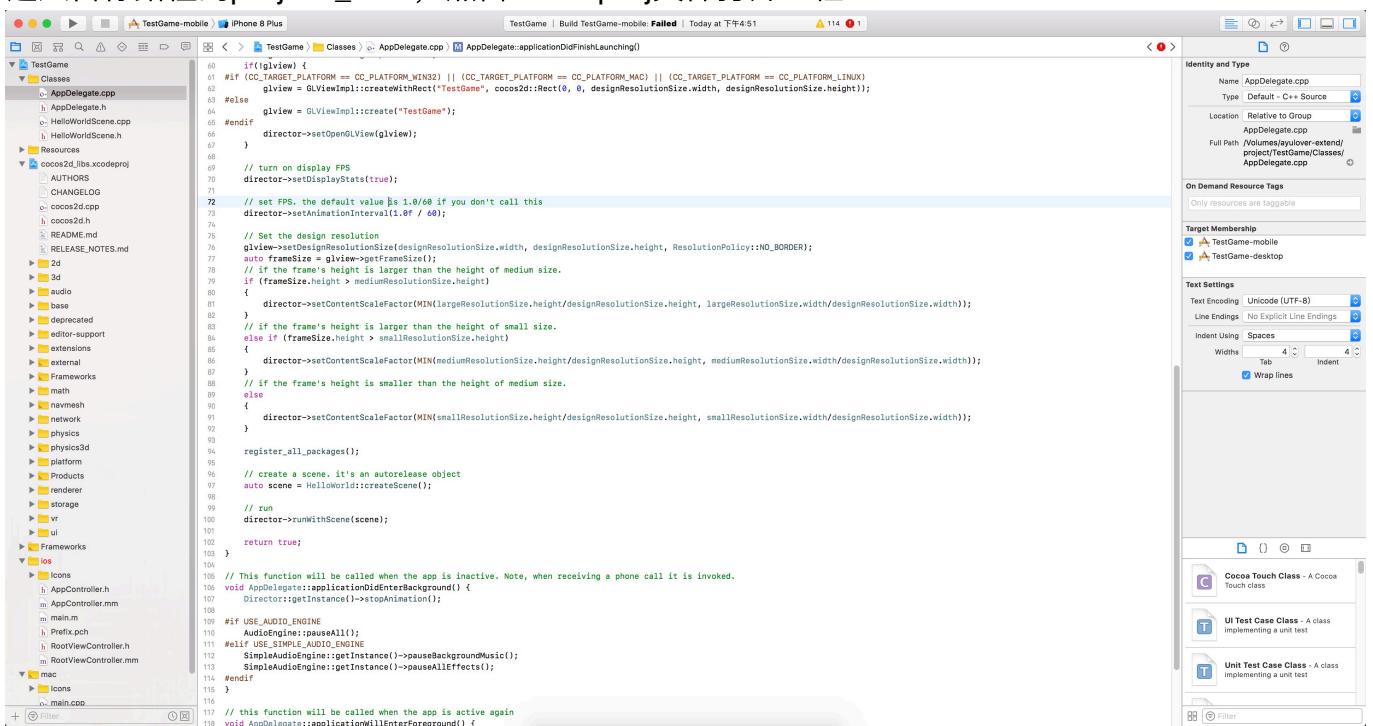
COCOS -V

2.2 初始化cocos工程

- 执行CLI初始化

```
cocos new [工程名] -p [工程标识名 (例: com.game)] -l [源语言(lua,js,cpp)] -d [目标路径]
```

- 进入目标路径的proj.ios_mac，点击xcodeproj文件打开工程



- 执行编译，点击play按钮



- 此时编译出错 (error:CCFileUtils.cpp->Call to unavailable function 'system': not available on iOS)

```

bool FileUtils::removeDirectory(const std::string& path)
{
    #if !defined(CC_TARGET_OS_TVOS)
        std::string command = "rm -r ";
        // Path may include space.
        command += "\"" + path + "\"";
        if (system(command.c_str()) >= 0)
            return true;
        else
            return false;
    #else
        return false;
    #endif
}

```

Call to unavailable function 'system': not available on iOS

原因：XCode9 将system API删除，使用ntfw API替换，解决方法如下

```

//在#include <dirent.h>下方添加
#if (CC_TARGET_PLATFORM != CC_PLATFORM_ANDROID)
#include <ftw.h>
#endif
//在removeDirectory方法上方添加
namespace
{
    #if (CC_TARGET_PLATFORM != CC_PLATFORM_ANDROID)
        int unlink_cb(const char *fpath, const struct stat *sb, int typeflag,
            struct FTW *ftwbuf)
        {
            int rv = remove(fpath);
            if (rv)
                perror(fpath);
            return rv;
        }
    #endif
}
//将removeDirectory方法替换为：
bool FileUtils::removeDirectory(const std::string& path)
{
    #if !defined(CC_TARGET_OS_TVOS)
    #if (CC_TARGET_PLATFORM != CC_PLATFORM_ANDROID)
        if (nftw(path.c_str(), unlink_cb, 64, FTW_DEPTH | FTW_PHYS) == -1)
            return false;
        else
            return true;
    #else
        std::string command = "rm -r ";
        // Path may include space.
        command += "\"" + path + "\"";
        if (system(command.c_str()) >= 0)
            return true;
        else
            return false;
    #endif
}

```

```
#endif // (CC_TARGET_PLATFORM != CC_PLATFORM_ANDROID)
#else
    return false;
#endif // !defined(CC_TARGET_OS_TVOS)
}
```

- 最终运行成功

