



“My 2187”  
安装部署说明书

Version 1.0



文件状态： [     ]    草稿 [ √ ]    正式发布 [     ]    正在修改	当前版本：	1.0
	作者：	李祖兴
	完成日期：	2016-07-15

## 文档修订摘要

日期	版本	说明	作者
2016-7-15	1.0	引言，系统配置，程序配置，运行和停止	李祖兴



<b>1</b>	<b>引言</b>	<b>4</b>
1.1	编写目的	4
1.2	背景	4
1.3	定义	错误！未定义书签。
1.4	参考资料	错误！未定义书签。
<b>2</b>	<b>系统配置</b>	<b>4</b>
2.1	运行环境	4
2.2	系统安装部署图	4
2.3	系统硬件配置	4
2.4	系统应用服务器软件安装与配置	4
<b>3</b>	<b>程序部署</b>	<b>5</b>
3.1	应用程序部署	5
3.1.1	参数配置	5
3.1.2	编译	11
3.1.3	打包	11
3.2	服务器参数设置	错误！未定义书签。
<b>4</b>	<b>运行和停止</b>	<b>12</b>

# 1 引言

## 1.1 编写目的

本说明书将对 My 2187 的安装与使用进行描述，使用户能够通过本说明书了解本软件的安装使用，以确保程序的正常运行。本文档的使用者是用户。

## 1.2 背景

根据王青老师发布的软件设计综合实验，我们小组合作开发出本游戏“My 2187”为作品用于参加本次软件设计综合实验。

# 2 系统配置

## 2.1 运行环境

- 操作系统使用 Windows 7、Windows 8/8.1、Windows 10 等 Windows 系统。
- 使用 git 作为代码管理工具。
- 使用 Visual Studio 2015 作为编译打包工具。
- 使用 cocos2dx 游戏开发引擎

## 2.2 系统安装部署

用户可以在 github 上面获取源码并编译打包（仓库地址下面给出），也可以直接下载我们的编译出来的程序，直接运行。

## 2.3 系统硬件配置

类别	标准配置	最低配置
计算机硬件	CPU: Pentium 4 或以上	CPU: Pentium 3
	内存: 1 GB 或以上	内存: 256 MB

## 2.4 系统应用服务器软件安装与配置

- Python 安装（请使用 2.x 版本）
- 获取 Cocos2dx 游戏引擎（请使用 3.x 版本）
- VS 安装

## 3 程序部署

本作品代码托管于 Github 上，仓库地址为：

<https://github.com/xingzuli/My2187>

编译完成的程序已放到百度云：

<https://pan.baidu.com/s/1jIAouFG>

### 3.1 应用程序部署

#### 3.1.1 参数配置

以下为 Cocos2dx 官网提供的配置环境方案：

-----分割线-----

## 工具准备

搭建开发环境需要安装工具包括

Visual Studio

python ——（本教程以 python2.7.3 版本为例），下载地址：

<http://www.python.org/download/releases/2.7.3/>。

Cocos2d-x ——（本教程以 cocos2d-x-3.0 版本为例），下载地址：

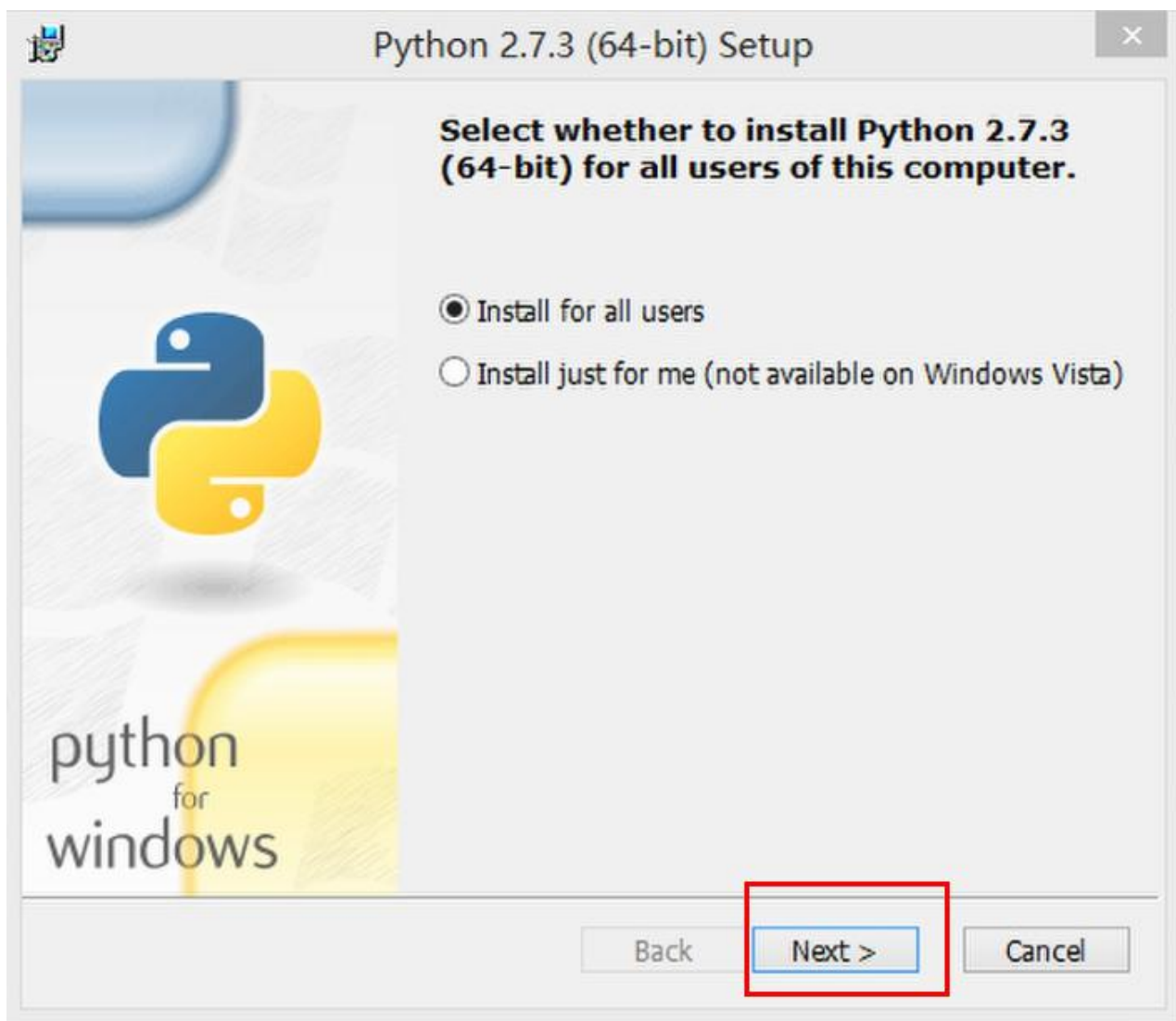
<https://code.google.com/p/cocos2d-x/downloads/list>。

## 安装配置过程

Visual Studio 的安装过程略过，本教程主要讲解 python2.7.3 和 cocos2d-x-3.2alpha0 的安装配置过程。

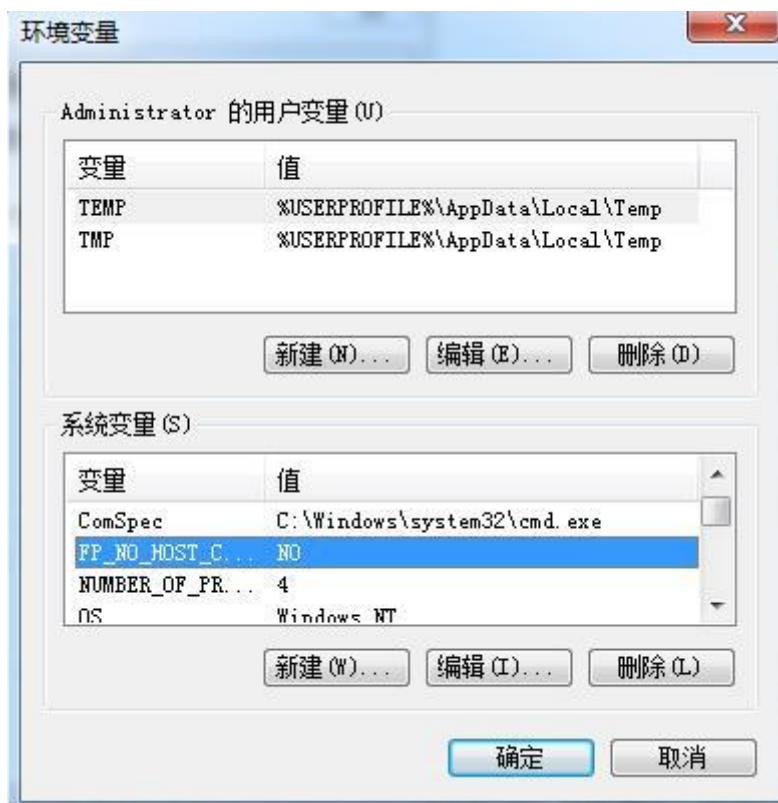
### 安装配置 python2.7.3

1. 下载并安装 python2.7.3。

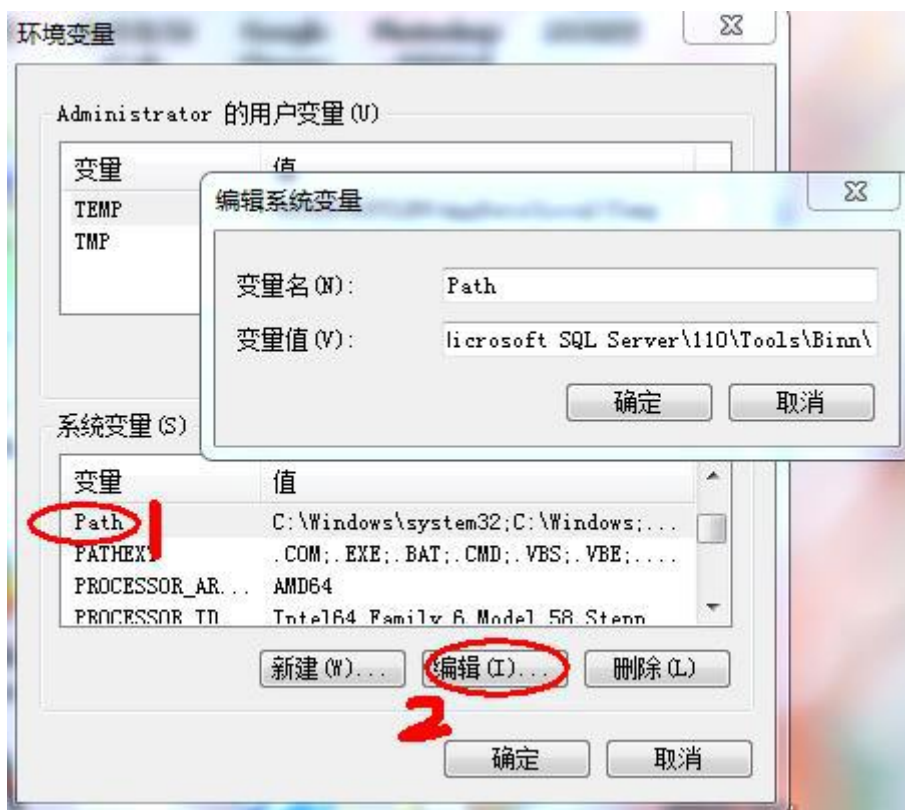


2. 配置环境

右键单击“计算机”，选择“属性”——》“高级系统设置”，在“高级”选项卡下选择“环境变量”！



在“系统变量”下选中“Path”，并点击下方的编辑按钮：



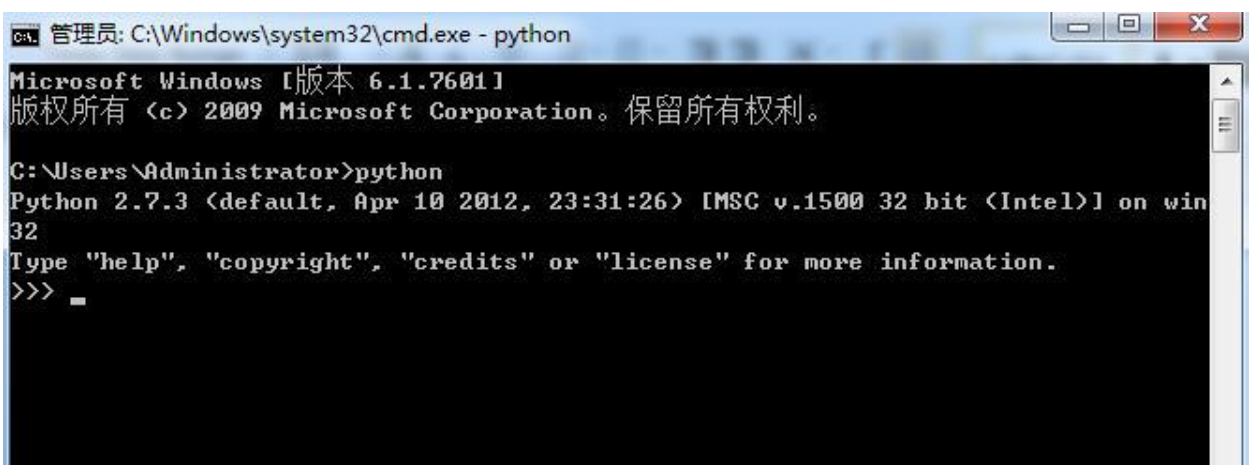
在弹出的“编辑系统变量”对话框中编辑“变量值”，在“变量值”的后面添加 Python 的安装路径（D:\Program Files (x86)\Python27），用英文分号（;）将其与后面的其他内容隔开。



同样地，在“PATHEXT”的变量值中添加 .PY;.PYM。

### 3. 检验安装是否成功

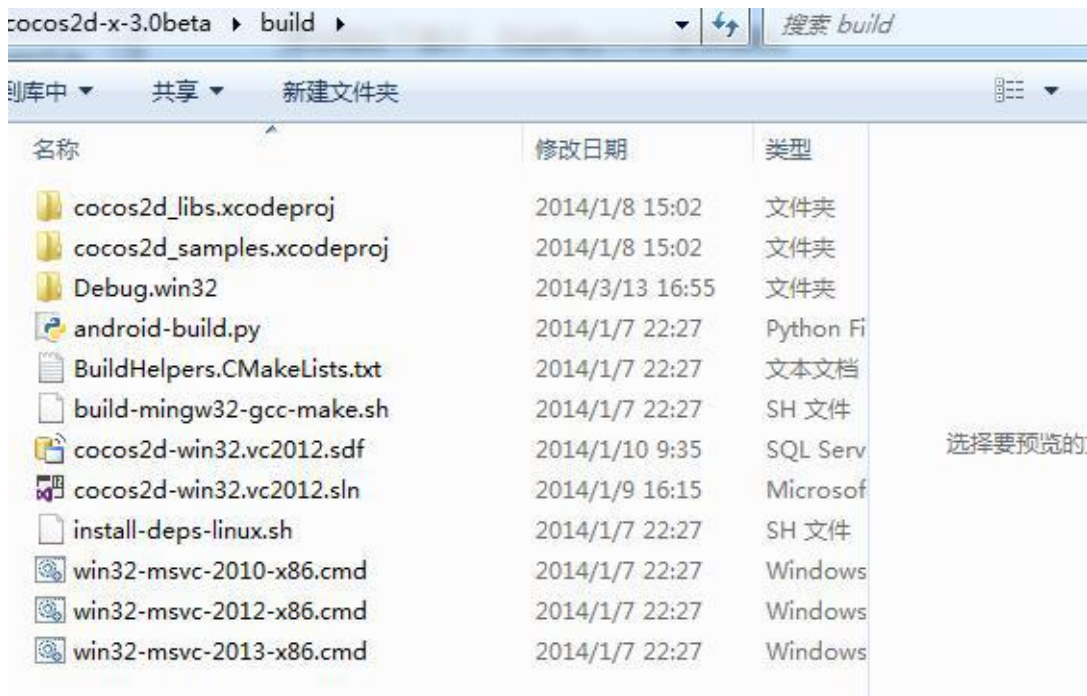
打开 cmd 控制台，输入 python，如果出现如下提示，则说明 python 安装成功。



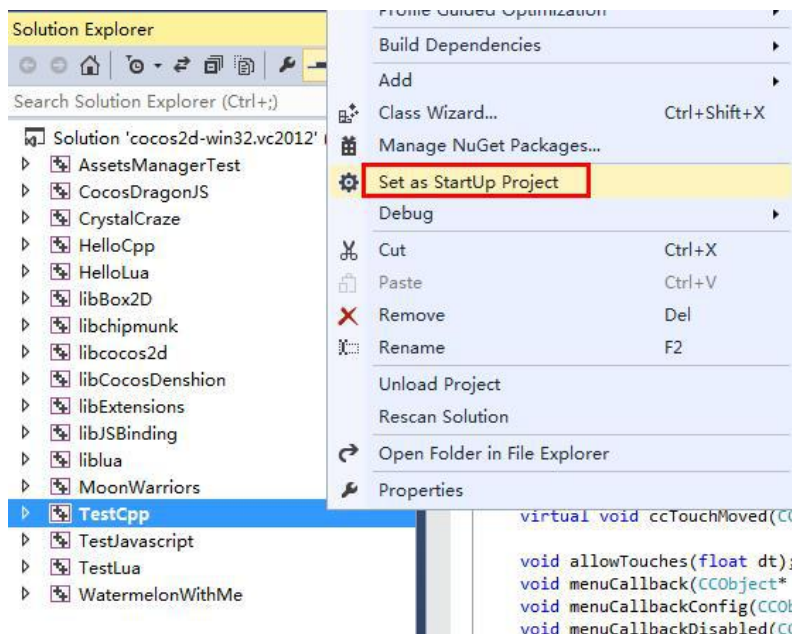


## 安装创建 cocos2d-x-3.2alpha0 项目

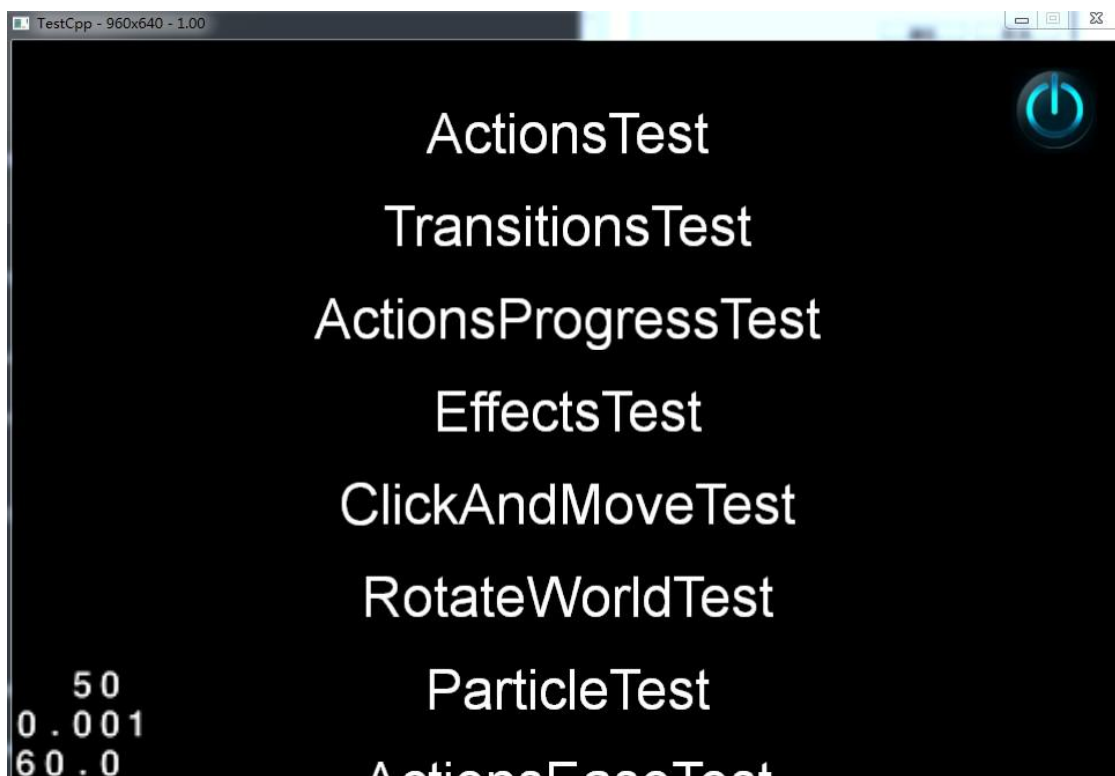
cocos2d-x-3.2alpha0 项目无需安装，下载 cocos2d-x-3.2alpha0 文件并解压，打开 “build” 目录，直接双击运行 cocos2d-win32.vc2012.sln 解决方案即可。



将你要运行的项目设为启动项，点击运行。



如果一切正常，你将得到如下的界面。那么祝贺你！你已成功地在 Windows 7 平台中运行 Cocos2d-x 引擎了。



### 3.1.1.1 创建项目

进入到目录 `cocos2d-x-3.2alpha0/tools/cocos2d-console/bin/cocos.py`

打开终端运行 `cocos.py` 脚本创建文件

```
python cocos.py new HelloCpp -p com.coco2dx.org -l cpp -d ~/Desktop
```

参数说明：

- HelloCpp 为项目名称
- `-p` 后面接包名
- `-l` 后面接开发语言类型，有 `cpp`，`lua`，`js` 三种类型
- 

`-d` 后面接项目存放的目录

- 
- 查看项目路径，你会发现目录中会出现新建的“HelloCpp”项目。依次打开“HelloCpp/proj.win32”文件夹，“HelloCpp.sln”解决方案，单击运行项目。出现下

面的界面则项目创建成功



-----分割线-----

使用以上方法创建项目成功后，从 github 上面获取我们的源码，把 src 里面的代码全部添加到项目的 Classes 目录(把原来4个文件清空)，把 resource 里的文件拷贝到项目的 Resource 文件夹中。

### 3.1.2 编译

做完上面的一切后，重新用 VS 打开项目，重新生成项目，使用 VS 编译运行即可。

### 3.1.3 打包

VS 默认为 Debug 模式，在此模式下生成的程序往往在其他人的机子上是跑不了的。我们需要做的有下面几步：

把工程调成 Release 模式。

右键项目->属性，打开项目属性页面，首先在配置属性->常规页面，找到“MFC 的使用”，选择“在静态库中使用 MFC”，然后在配置属性->C/C++->代码生成页面，在“运行时库”里面选“多线程 (/MT)”

然后重新生成项目，编译运行。然后我们可以看到多了个 Release.win32 文件夹，把里面的 dll 库和 exe 程序以及所用到的资源 (\*.png, \*.Mp3, \*.jpg 等等)提取到另外文件夹。

把该文件夹压缩成 rar 格式，使用 WinRAR 自解压打包成应用程序。

## 4 运行和停止

打包成 exe 程序后，在 Windows 平台可以直接运行起来了。



进入游戏界面，像普通程序一样运行即可。