

# 现代操作系统应用开发实验报告

学号： 15331046

班级： 软工一班

姓名： 陈志扬

实验名称： HelloCocos

## 一. 参考资料

请在这里列出对本实验有帮助你所参考的资料或者网站。

资料：老师提供的课件、TA 师兄的作业 ppt，github 上 cocos2d-x 源码

网站：cocos2d-x 源码 <https://github.com/cocos2d/cocos2d-x>

解决中文乱码的参考博客 <http://www.cnblogs.com/msxh/p/5588775.html>

官方 api <http://api.cocos.com/> 等等

## 二. 实验步骤

请在这里简要写下你的实验过程。

① 根据师兄给的作业 ppt 指导，安装 python2.7 ( 我安装的版本是 **python2.7.13** )

和 cocos2d-x ( 我安装的版本是 **cocos2d-x-3.14.1** )，并配置 python 环境，出现

如下结果，说明安装正确。

```
C:\Users\linji>python
Python 2.7.13 (v2.7.13:a06454blafal, Dec 17 2016, 20:42:59) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

② 然后在 cocos2d-x 下载目录中打开 cmd 执行命令 **python setup.py**，安装完成后

输入 **cocos** 可见到以下结果说明安装正确。

```
C:\Users\linji>cocos
D:\Cocos\cocos2d-x-3.14.1\tools\cocos2d-console\bin\cocos.py 2.2 - cocos console: cocos2d-x 的命令行工具集。

可用的命令:
run          在设备或者模拟器上编译，部署和运行工程。
gen-libs     生成引擎的预编译库。生成的库文件会保存在引擎根目录的 'prebuilt' 文件夹。
luacompiled 对 lua 文件进行加密和编译为字节码的处理。
deploy       编译并在设备或模拟器上部署工程。
package      管理 cocos 中的 package。
compile       编译并打包工程。
gen-simulator 生成 Cocos 模拟器。
new           创建一个新的工程。
jscompile     对 js 文件进行加密和压缩处理。
gen-templates 生成用于 Cocos Framework 环境的模板。

可用的参数:
-h, --help      显示帮助信息。
-v, --version    显示命令行工具的版本号。
--ol ['en', 'zh', 'zh_tr'] 指定输出信息的语言。

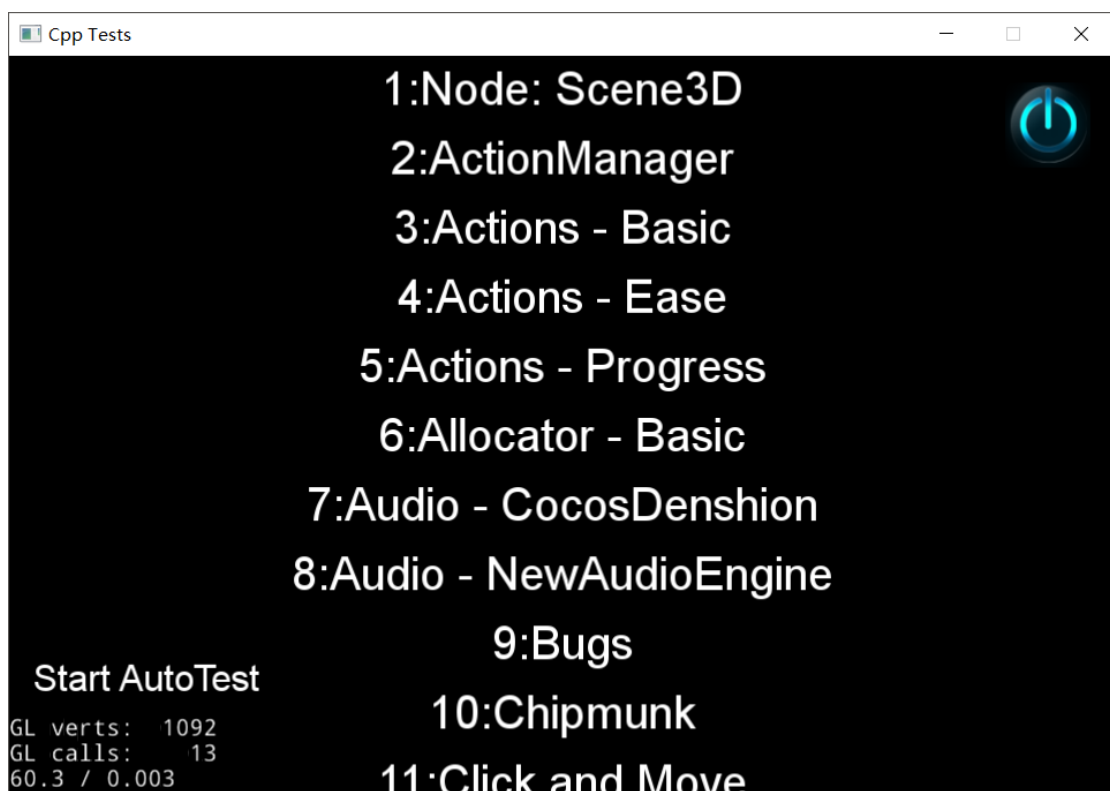
示例:
cocos new --help
cocos run --help
```

③ 然后我在下图所示的文件夹中**创建一个新项目 HelloCocos** ,并打开 proj.win32 中

的项目，用 vs2015 编译运行，然后仔细阅读代码，了解其运行流程。



④ 接下来**编译运行源码中的 cpp-tests**，打开源码中的 **build** 文件夹，用 vs2015 打**开 cocos2d-win32.sln**，将 **cpp-tests** 设为启动项，编译运行。可得到如下结果：

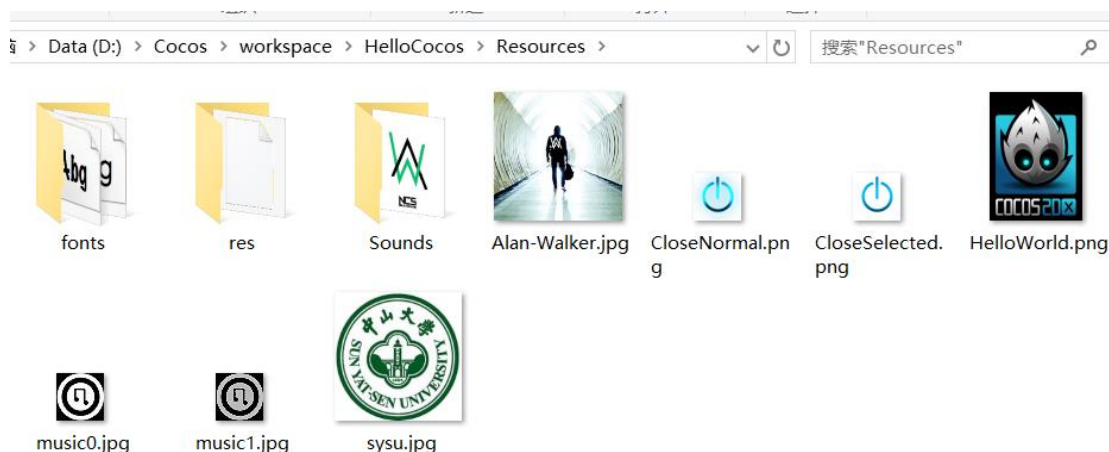


⑤ 至此，前面的准备工作均完成，可开始本次实验了。先做**更换背景图片**这部分，只

需要把一张图片放到 Resource 文件夹中即可，相应的代码修改一下即可。

- ⑥ 接下来，添加一个 label 来显示姓名，因为原来已经有一个 label，我们可以修改一下位置即可用来显示学号。
- ⑦ 创建一个 xml 文件，里面包含姓名和学号，接下来查找相关资料解决 cocos2d-x 中文乱码的问题，因此这一部分也迎刃而解。
- ⑧ 然后，添加一个 MenuItem，并为之添加简单的触发事件，这里我设置了一个背景音乐，并更换了原图片。
- ⑨ 最后，为学号姓名 label 添加一些样式，如字体和颜色。


这里给出 Resource 文件夹中的文件截图：图片新增了 Alan-Walker.jpg、sysu.jpg、music0.jpg 和 music1.jpg



Fonts 文件夹中添加了两个字体：斜体 LHANDW.TTF 和 simkai.ttf

Data (D:) > Cocos > workspace > HelloCocos > Resources > fonts				搜索"fonts"
名称	修改日期	类型	大小	
arial.ttf	2017/4/20 23:20	TrueType 字体文件	761 KB	
LHANDW.TTF	2015/7/30 3:27	TrueType 字体文件	62 KB	
Marker Felt.ttf	2017/4/20 23:20	TrueType 字体文件	26 KB	
simkai.ttf	2017/2/11 2:08	TrueType 字体文件	11,511 KB	

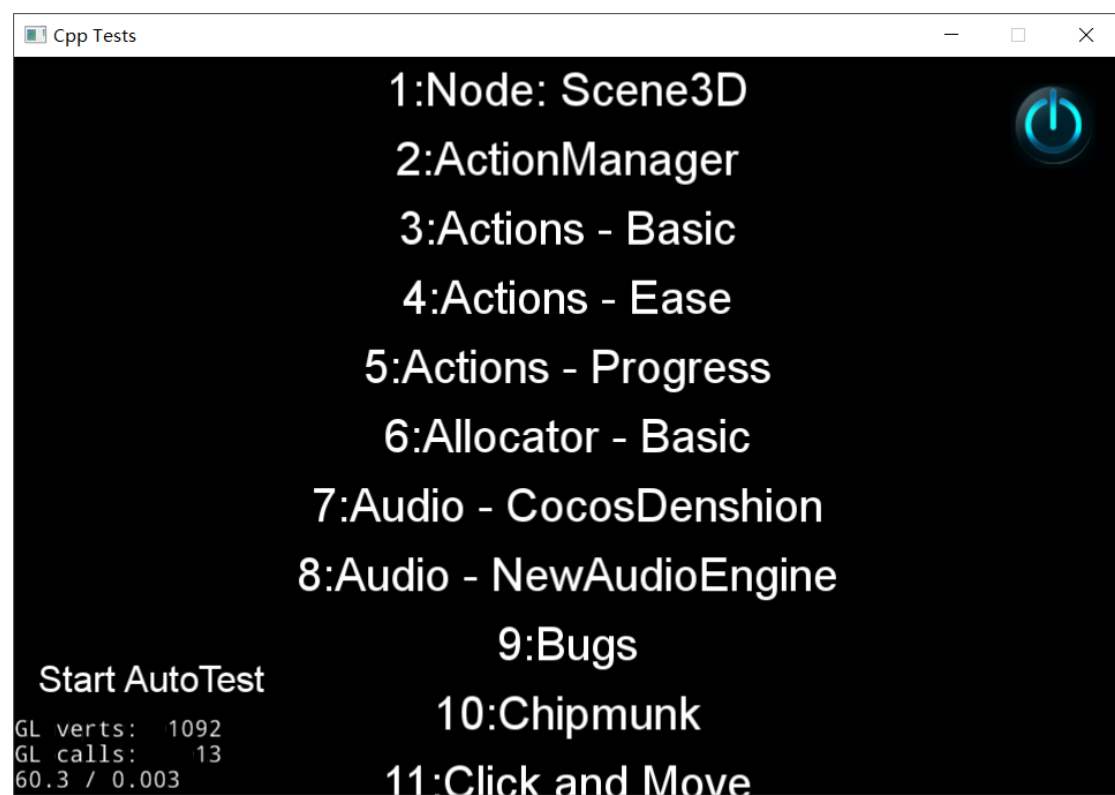
Sounds 文件中包含一首歌，用来做背景音乐：

(D:) > Cocos > workspace > HelloCocos > Resources > Sounds				
名称	#	标题	参与创作的艺术家的	唱片集
 Fade.mp3	1	Fade	Alan Walker	Fade

### 三 . 实验结果截图

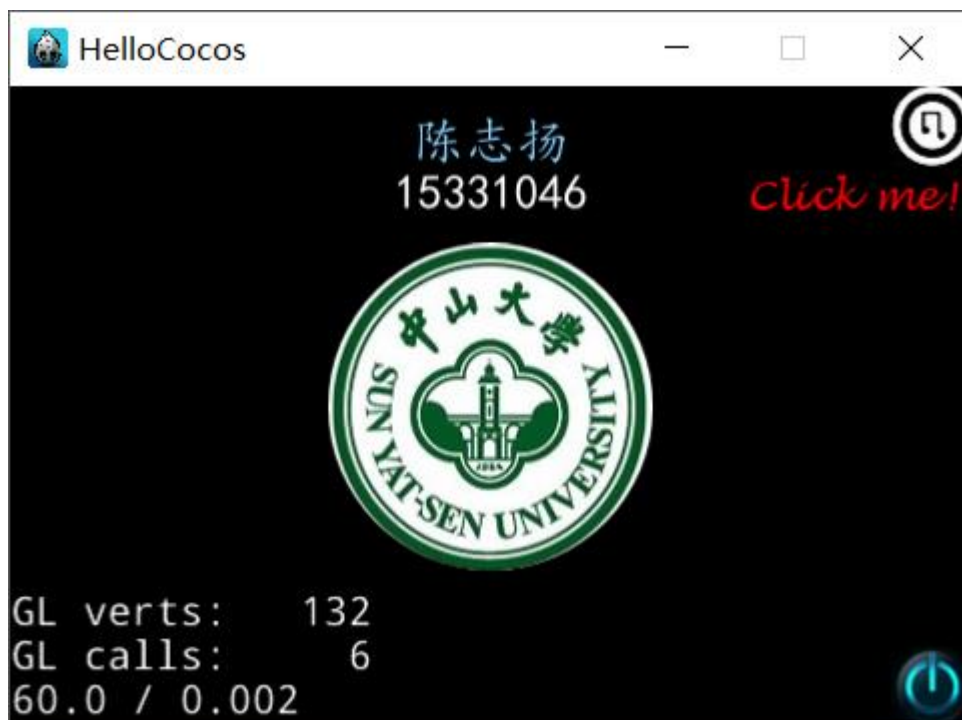
请在这里把实验所得的运行结果截图。

Cpp-tests 运行后的截图：





实验 HelloCocos 编译运行后的结果：可以看到学号、姓名、背景图片，除此之外，右上角添加一个 **MenuItem**，当点击它可播放音乐，此时背景图片已更换。**Click me!** 为斜体红色，学号姓名均为楷体，姓名也加上亮蓝色。因此，该实验结果表明已基本实现了实验要求。



点击右上角可播放音乐：



#### 四．实验过程遇到的问题

请在这里写下你在实验过程中遇到的问题以及解决方案。

- ① 主要的问题还是中文乱码的问题，网上这方面的博客很多，可是很多基本没说出造成乱码的关键问题。所以一开始我按照参考博客的做法，可是依旧解析不出中文，后来发现不要用 `createWithTTF` 这个函数，直接改成 `create` 函数就可以解决该问题了，对此我有点疑问，所以我上网查了一下，原来 `fonts` 文件夹中 `arial.ttf` 和 `Marker Felt.ttf` 根本不支持中文，就算改成 `create` 函数也依然行不通。

参考截图：

```
auto *chnStrings = Dictionary::createWithContentsOfFile("myName.xml");
const char *name = ((String*)chnStrings->objectForKey("myName"))->getCString();

// this is also available
auto nameLabel = Label::create(name, "fonts/arial.ttf", 26);
//auto nameLabel = Label::createWithTTF(name, "fonts/simkai.ttf", 26);
```

亦或是：

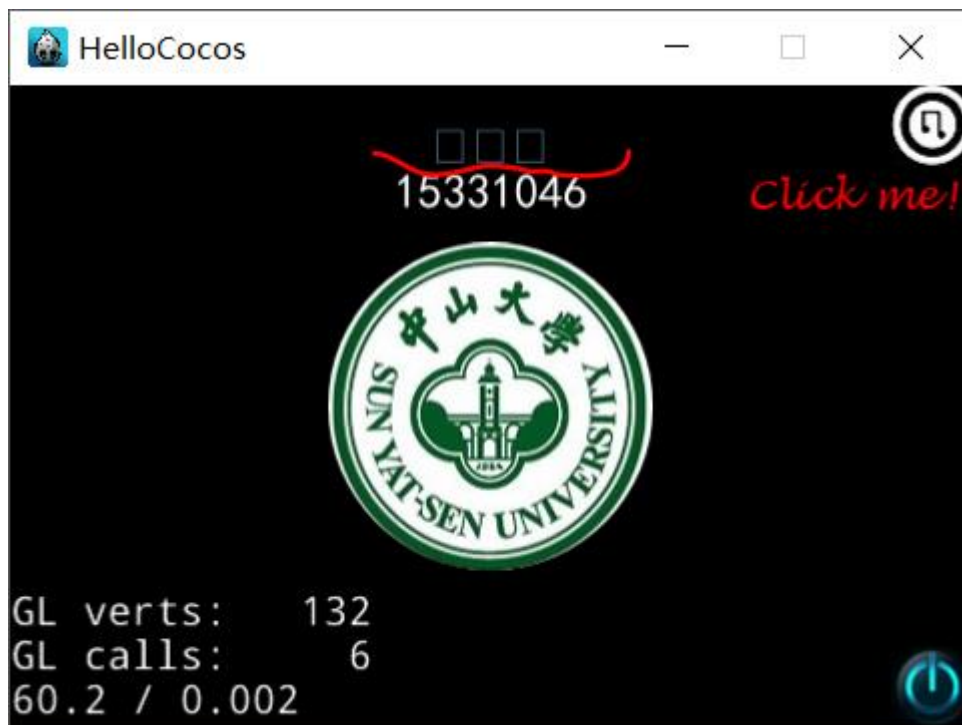
```

auto *chnStrings = Dictionary::createWithContentsOfFile("myName.xml");
const char *name = ((String*)chnStrings->objectForKey("myName"))->getCString();

// this is also available
auto nameLabel = Label::create(name, "fonts/Marker Felt.ttf", 26);
//auto nameLabel = Label::createWithTTF(name, "fonts/simkai.ttf", 26);

```

这两种不同的代码的运行结果：**中文乱码**



而如果我们用下面这种字体，同样用上 create 函数：

```

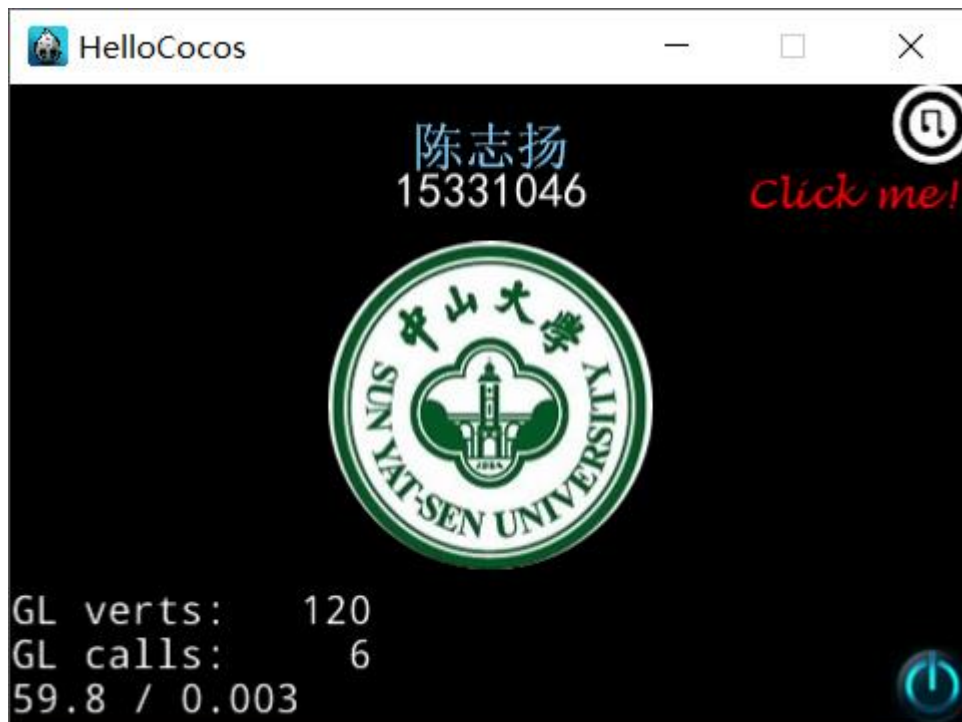
auto *chnStrings = Dictionary::createWithContentsOfFile("myName.xml");
const char *name = ((String*)chnStrings->objectForKey("myName"))->getCString();

// this is also available
auto nameLabel = Label::create(name, "Arial", 26);
//auto nameLabel = Label::createWithTTF(name, "fonts/simkai.ttf", 26);

```

运行结果即可正常显示中文，原因就在于 Arial 支持中文显示了：

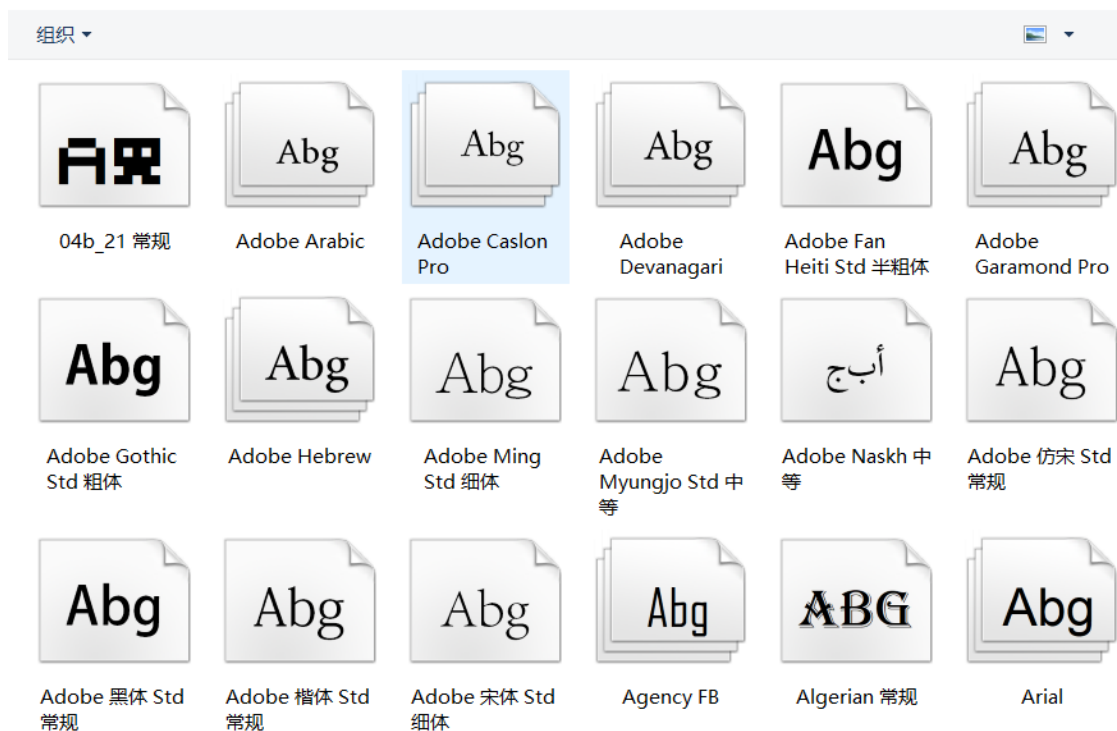




弄清楚这个问题后，我还是用了 `createWithTTF` 函数，在系统盘 c 盘中拷贝字体到 `fonts` 文件夹中，只需要在代码中修改相应的文件名即可。

OS (C:) > Windows > Fonts >

[预览、删除或者显示和隐藏计算机上安装的字体](#)



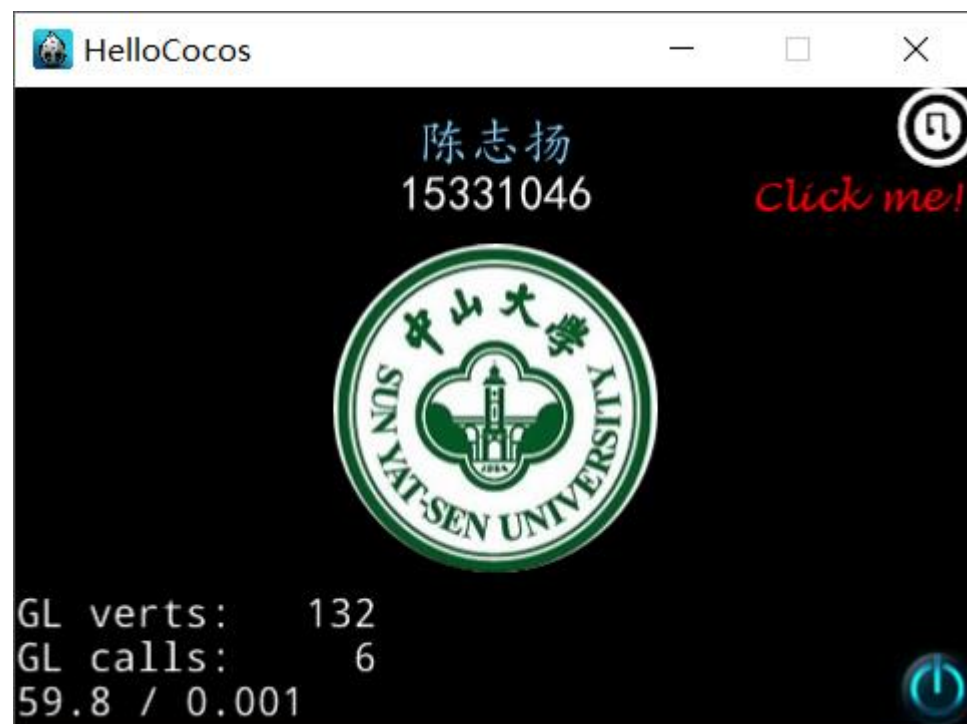


代码如下：

```
auto *chnStrings = Dictionary::createWithContentsOfFile("myName.xml");
const char *name = ((String*)chnStrings->objectForKey("myName"))->getCString();

// this is also available
// auto nameLabel = Label::create(name, "Arial", 26);
auto nameLabel = Label::createWithTTF(name, "fonts/simkai.ttf", 26);
```

运行结果：



- ② 添加一个 MenuItem，主要处理位置和音乐播放的问题。由于对位置的处理还不是非常熟悉，所以有点盲人摸象的感觉，希望接下来可以更好解决该问题。音乐主要查找相关 api 即可获得。

具体的代码如下：

```
#include "HelloWorldScene.h"
#include "SimpleAudioEngine.h"

USING_NS_CC;
using namespace CocosDenshion;
```

添加一个 music 的 menu:

```

// add a "music" icon to play the music
auto musicItem = MenuItemImage::create(
    "music0.jpg",
    "music1.jpg",
    CC_CALLBACK_1(HelloWorld::click, this));

musicItem->setPosition(Vec2(origin.x + visibleSize.width - musicItem->getContentSize().width / 2,
    origin.y + musicItem->getContentSize().height / 2));

// create menu, it's an autorelease object
menu = Menu::create(musicItem, NULL);
menu->setPosition(Vec2(0, 280));
this->addChild(menu, 1);

```

添加一个提示“Click me!”的 label:

```

// add a click label shows "Click me!"
auto label = Label::createWithTTF("Click me!", "fonts/LHANDW.TTF", 20);
// the position of the label
label->setPosition(Vec2(origin.x + visibleSize.width - musicItem->getContentSize().width * 1.5,
    origin.y + visibleSize.height - label->getContentSize().height * 2));
// the color of the label
label->setColor(ccc3(255, 0, 0));
this->addChild(label, 1);

```

Click 事件:

```

void HelloWorld::click(cocos2d::Ref * pSender)
{
    if (!isClicked) {
        auto visibleSize = Director::getInstance()->getVisibleSize();
        Vec2 origin = Director::getInstance()->getVisibleOrigin();

        // add a sprite
        auto sprite = Sprite::create("Alan-Walker.jpg");
        // position the sprite on the center of the screen
        sprite->setPosition(Vec2(visibleSize.width / 2 + origin.x, visibleSize.height / 2 + origin.y));
        this->addChild(sprite, 0);

        // play the music
        SimpleAudioEngine::getInstance()->playBackgroundMusic("Sounds/Fade.mp3", true);
        isClicked = true;
    }
}

```

## 五 . 思考与总结

请在这里写下你本次试验的心得体会以及所思所想。

这次实验要求有学号、姓名、背景图片，并为相应的 label 设置文字样式，添加一个 MenuItem 并为之添加一个触发事件。针对文字样式，我主要改变的文字的颜色和字体形式；对于 MenuItem 我添加了一个点击可播放音乐的触发事件，基本实现了实验要求。本次实验的实验内容比较简单，所以很快就能完成，我们学习的

要点主要是熟悉 cocos2d-x 的基础概念 “渲染框架”：由精灵、层、场景、导演、等节点构成的树形渲染结构来进行图形的绘制功能，为接下来的实验打好坚实的基础。

1. 实验报告提交格式为 pdf。
2. 实验内容不允许抄袭，我们要进行代码相似度对比。如发现抄袭，按 0 分处理。