**实验报告1：C#控制台编程**

曾洋 3017218147

1. **实验要求**

使用C#控制台编程实现简单的二维码生成器，支持直接在控制台打印和批量保存二维码图片文件。

1. **实验内容**

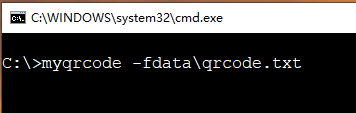
a. 当用户在命令行输入参数时，判断参数是否是QrCode可以生成的信息，长度是否符合本程序的基本要求（自定义长度），如果符合要求就直接产生QrCode的结果，将QrCode的编码矩阵在控制台屏幕上输出对应黑、白字符方块组成的QrCode码。

b. 如果不符合要求，提示用户输入正确的命令行信息，运行结束。

c. 将程序的每一行都加上注释，并说明其功能。

d. 用手机识别是否可以还原数据，如果不能，请分析原因，修改程序，直至手机能正确识别QrCode内的原有信息。

e. 在命令行里传递一个文件名(包括文件所在目录，可以是可执行文件所在目录的相对目录例如data\qrcode.txt)，该文件是文本文件，文件中每行有一条字符串信息。假定myqrcode为本程序编译后的可执行文件名，-f表示QrCode信息放在-f后的data\qrcode.txt文件中。在控制台界面输入如下命令，可以将qrcode.txt中的每一行信息生成一个QrCode。

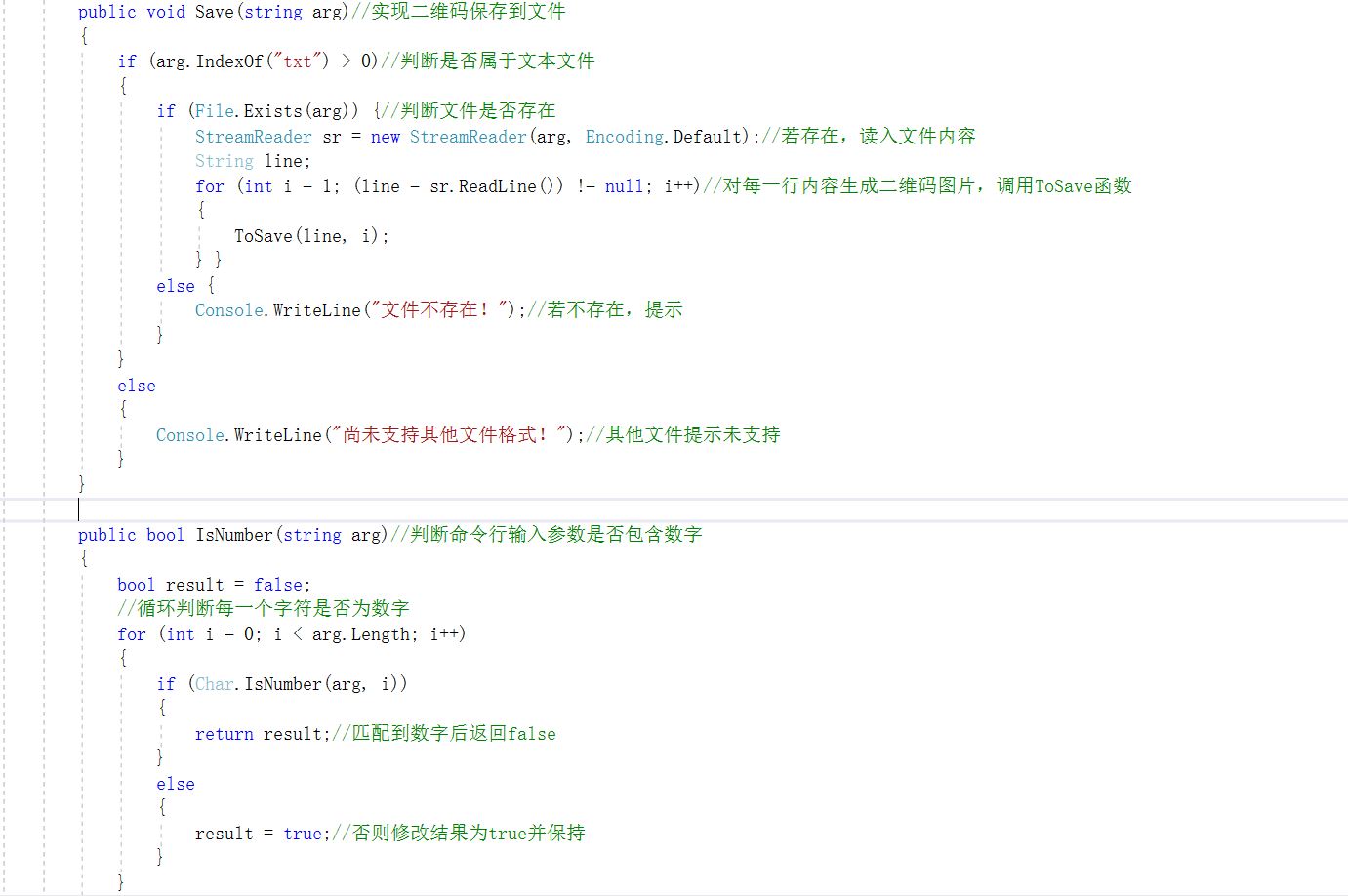


生成的QrCode需要保存到.png或bmp文件中，保存的文件名以信息所在行号三位数+信息的前四个字符构成，例如:”001+测试QR.png“为该文件的第一行信息生成的QrCode图片。

如果没有-f则以上述a的方式在控制台输出QrCode。此应用场景用于大量快速生成QrCode图片的应用场合。例如：支付宝卖支付码贴纸的场景。此外，如果QrCode信息是记录在Excel文件里或MySql数据库里，是否可以实现同样的功能。三种文件类型都能实现则更好。需要自己调研的内容：C#.NET的文本文件打开/关闭/读取；Office的Excel文件的打开/读取；数据库课程中的MySql数据库的表的建立/读取,.NET与MySql的连接方式。

1. **详细设计**

实验详细代码如下：



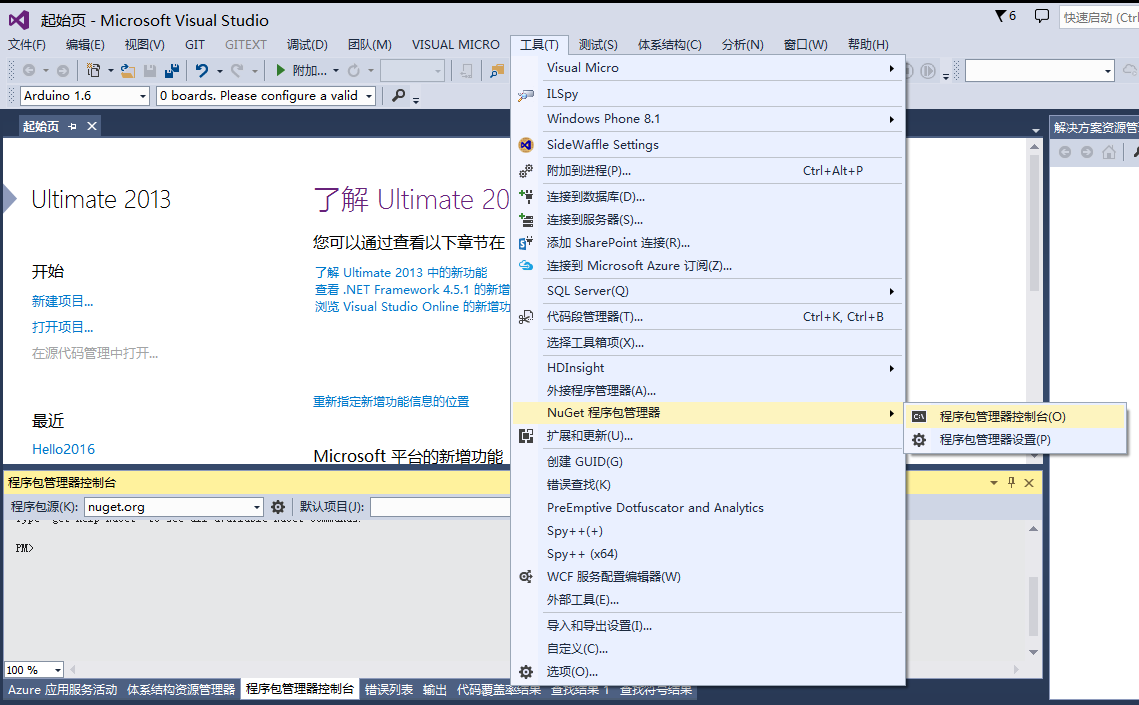
1. **上机实验步骤**

1) 新建一个控制台项目

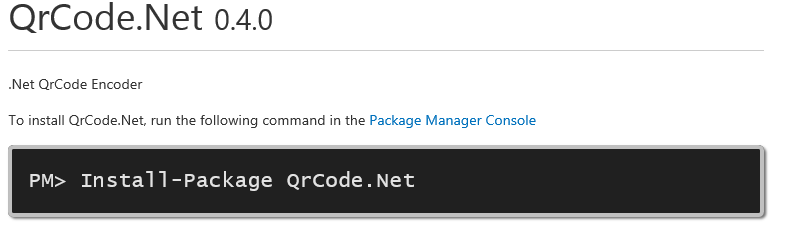
2) 安装QrCode库：

a. 第一种安装方式

1.如下图方式，菜单“工具”🡺“NuGet程序包管理器”🡺 点击“程序包管理器控制台”



2.在“程序包管理器控制台”中输入“Install-Package QrCode.Net”



3.正常安装结果如下：

PM> Install-Package QrCode.Net

正在安装“QrCode.Net 0.4.0.0”。

已成功安装“QrCode.Net 0.4.0.0”。

正在将“QrCode.Net 0.4.0.0”添加到 QrCodeTest。

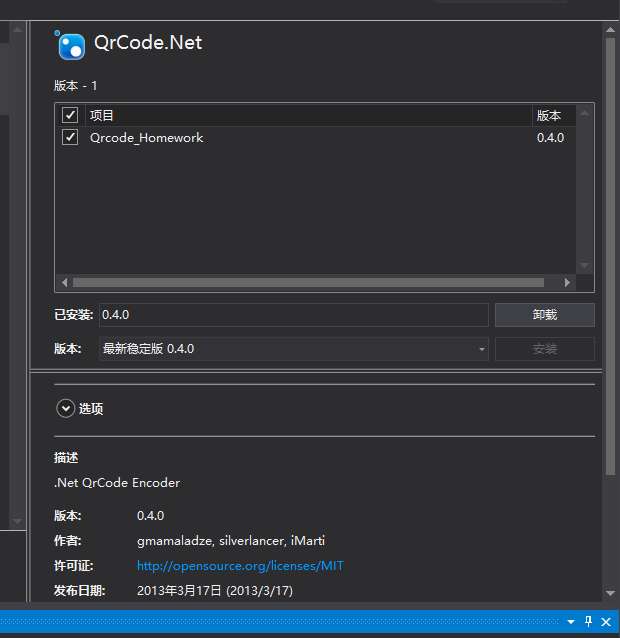
已成功将“QrCode.Net 0.4.0.0”添加到 QrCodeTest。

b. 第二种安装方式

1.在工具中找到NuGet包管理器--->管理解决方案的NuGet程序包--->输入QrCode.Net进行搜索



2.点击安装即可



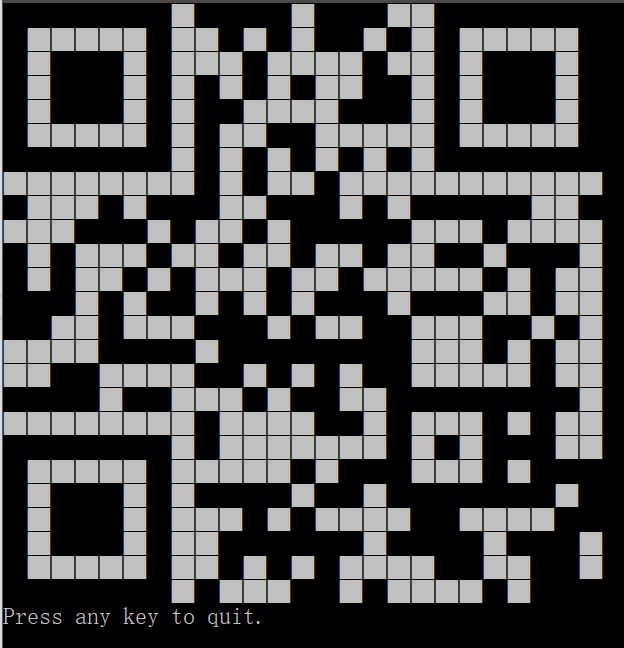
3) 程序编写实现“实验内容”中要求的内容。

1. **结论**

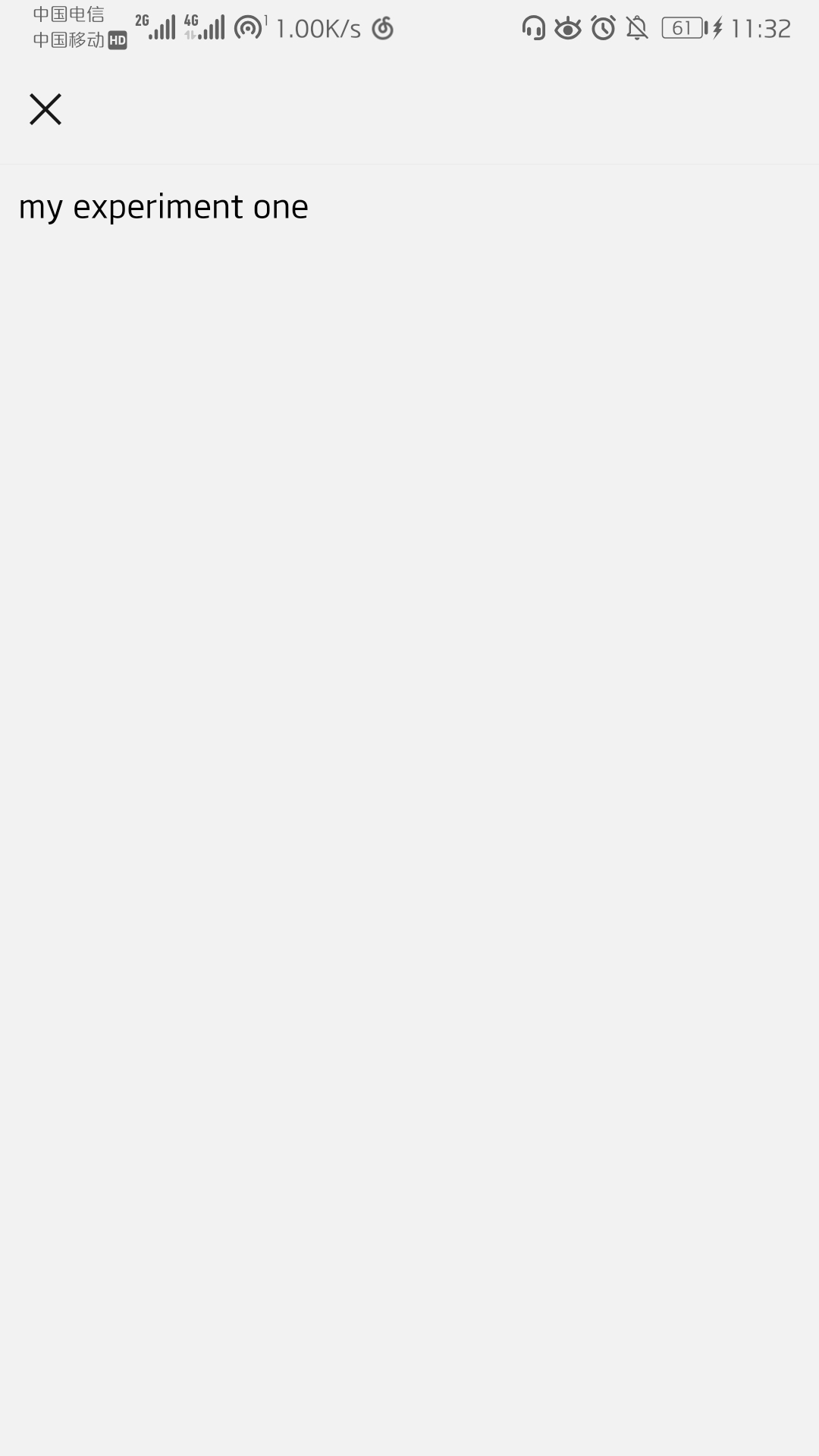
**1. 实现在控制台打印二维码**

在命令行输入参数“my experiment one"

运行结果如下：



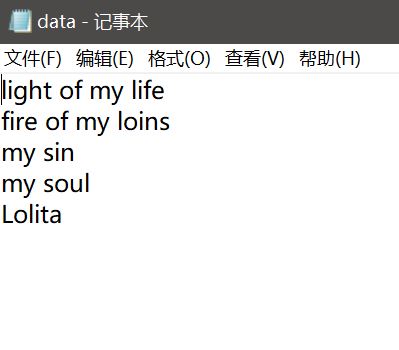
使用手机扫码得到结果如下：



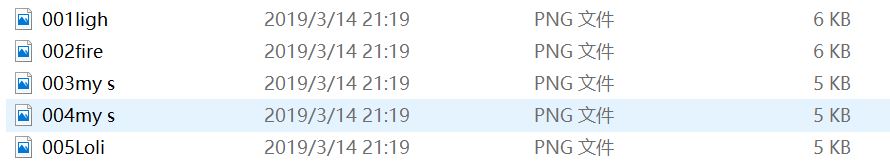
**2. 实现从文本文件读入二维码并批量保存图片**

在命令行输入参数”-f data.txt"

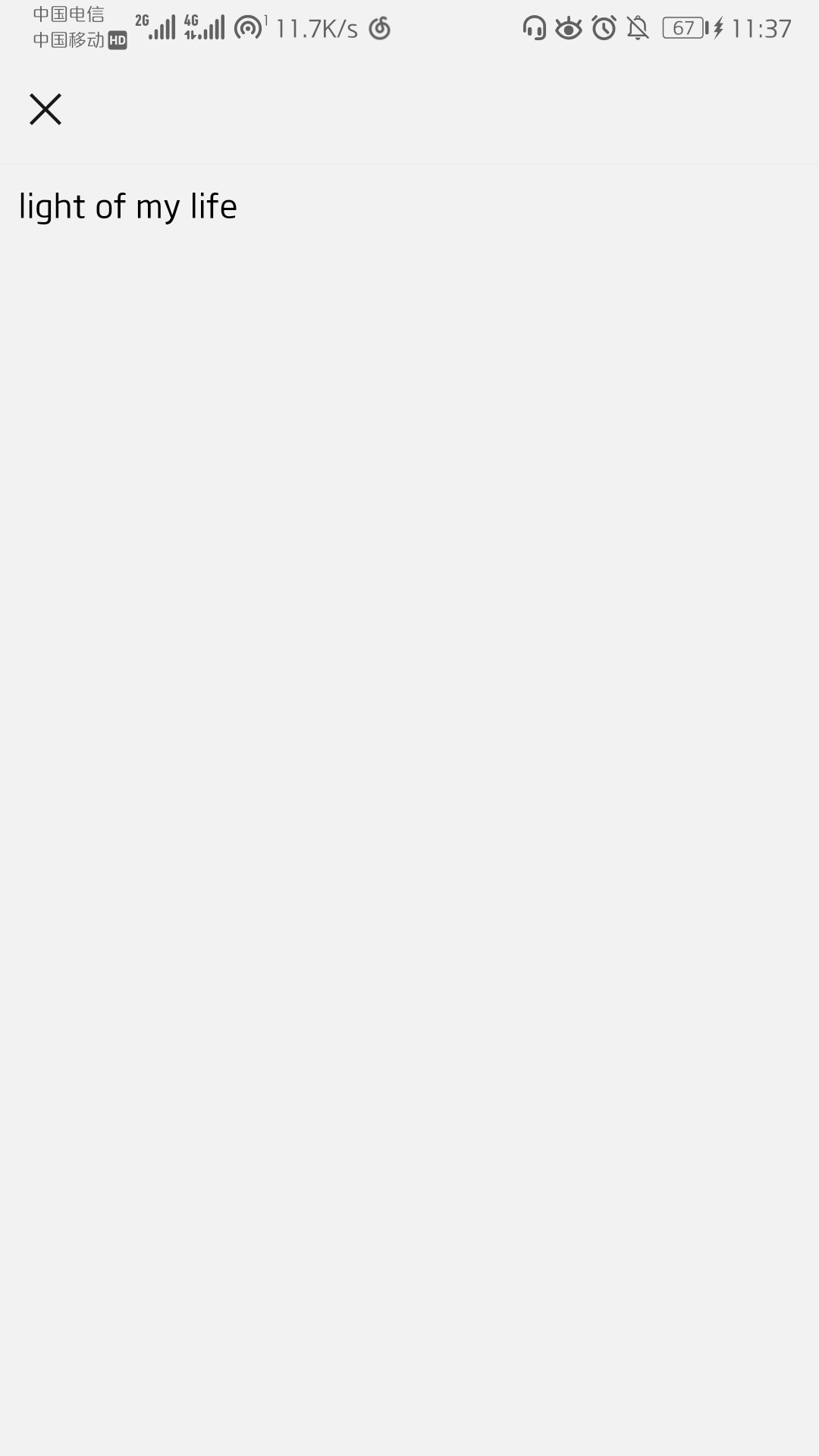
其中data.txt是一个保存有待生成二维码信息的文本文件，其内容如下：



程序运行后将每一行信息分别生成二维码保存到指定位置：



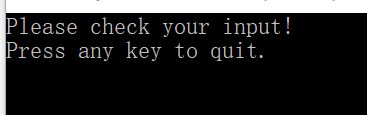
使用手机扫码得到部分结果如下：



**3. 对异常情况的处理**

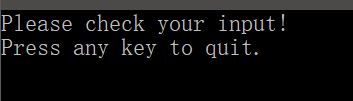
**a) 无参数输入的情况**

提示检查输入：



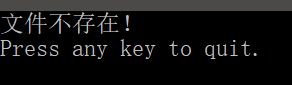
**b) 无文件地址的情况**

提示检查输入：



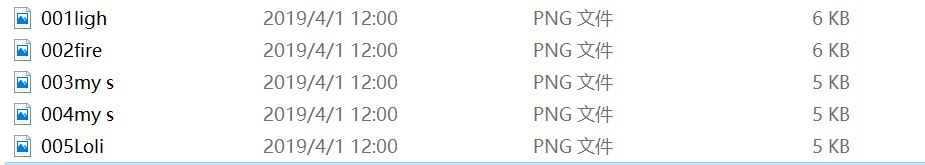
**c) 文件不存在的情况**

提示文件不存在：



**d) 第二次执行正确命令时覆盖原文件**

成功覆盖：



**e) 参数超出规定长度的情况**

提示参数不合法：

