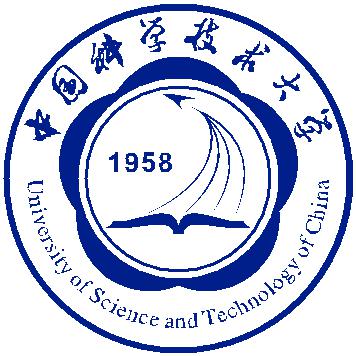


本科实验报告

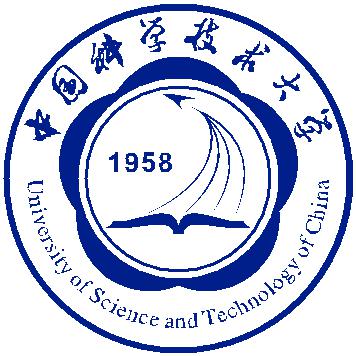


英语字母频率分析

|  |  |  |  |
| --- | --- | --- | --- |
| 作者姓名： |  | 吴立凡 | |
|  |  |  |  |
| 学科专业： | 网络空间安全 | | |
|  |  |  |  |
| 导师姓名： |  | 李卫海 | |
|  |  |  |  |
| 完成时间： |  | 2020年5月 | |
|  |  |  |  |

University of Science and Technology of China

A research report for bachelor’s degree



English letter frequency analysis

|  |  |
| --- | --- |
| Author : | Lifan Wu |
| Speciality : | Cyberspace Security |
| Supervisor : | Weihai Li |
| Finished Time : | May, 2020 |

致谢

感谢CSDN的前辈们，他们的论文博客为本次实验提供了丰富的经验与建议。如果不是他们的帮助，这篇文章的完成也会变得无比艰难。

另外也要感谢所有教过我的老师们，他们所教授的许多内容都成为了我实践的坚实基础。在此特别感谢李卫海老师，他所教授的密码学导论为这篇论文提供了莫大的帮助。

目录

[摘要 5](#_Toc72873132)

[关键词 6](#_Toc72873133)

[Abstract 6](#_Toc72873134)

[Keywords: 6](#_Toc72873135)

[正文 7](#_Toc72873136)

[题目要求 7](#_Toc72873137)

[算法分析 7](#_Toc72873138)

[完成效果 8](#_Toc72873139)

[分析《圣经》中字母统计分布 9](#_Toc72873140)

[三字符的可视化处理 12](#_Toc72873141)

[用维吉尼亚密码加密这本书 15](#_Toc72873142)

[使用kasiski方法破解vigenere密码 17](#_Toc72873143)

[参考文献 19](#_Toc72873144)

# 摘要

对于代换密码，统计分析是一种很有效的攻击方式，这是因为简单代换密码加密后的结果往往有很多统计特征，给攻击方许多攻击的机会。本次研究开发了一个用c ++语言和python开发的文字统计分析工具，用于统计一段文字中单字符、双字符、三字符的出现频率，并使用图表表示出来。然后使用简单的代换密码：维吉尼亚密码对《圣经》进行加密，对密文进行统计分析，并用Kasiski方法分析密钥长度。

# 关键词

字母 频率 双字符 密度图 维吉尼亚密码 Kasiski方法

# Abstract

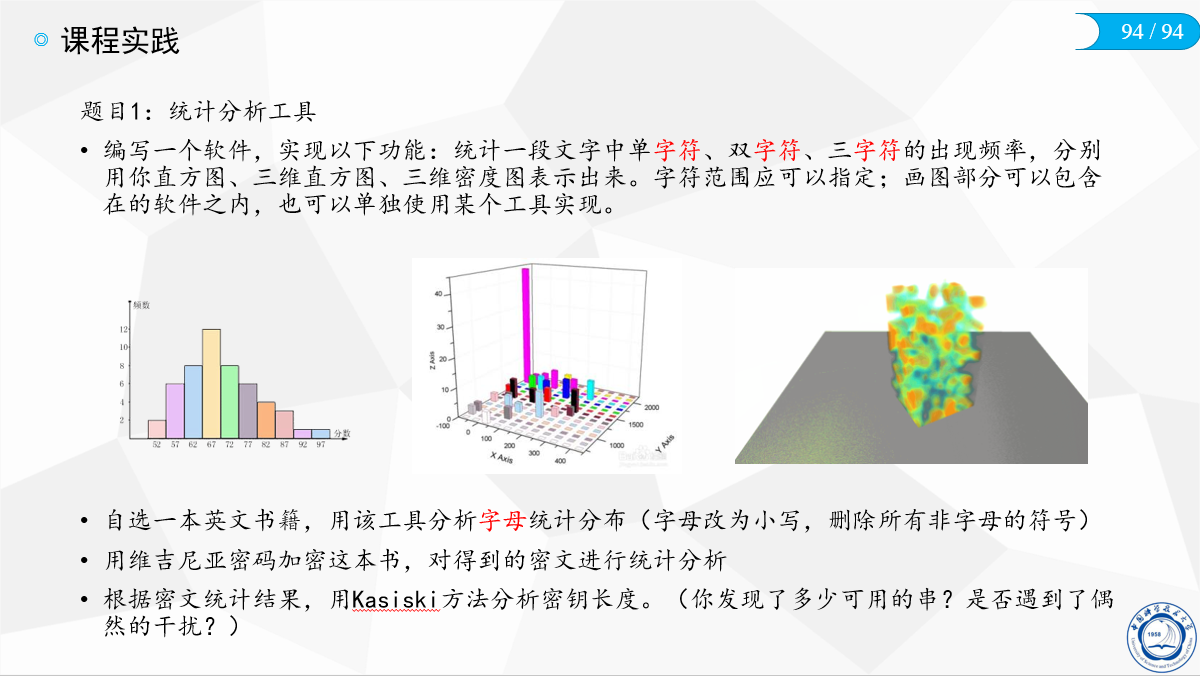
For substitution passwords, statistical analysis is a very effective attack method. This is because the result of simple substitution password encryption often has many statistical characteristics, giving the attacker many opportunities for attack. In this research, a text statistical analysis tool developed in c++ language and python was developed to count the frequency of single, double, and three characters in a text, and use graphs to show it. Then use a simple substitution password: the Virginia cipher encrypts the "Bible", performs statistical analysis on the ciphertext, and uses the Kasiski method to analyze the key length.

# Keywords:

Letter frequency two-character density map Virginia cipher Kasiski method

# 正文

## 题目要求



使用语言：c语言、python

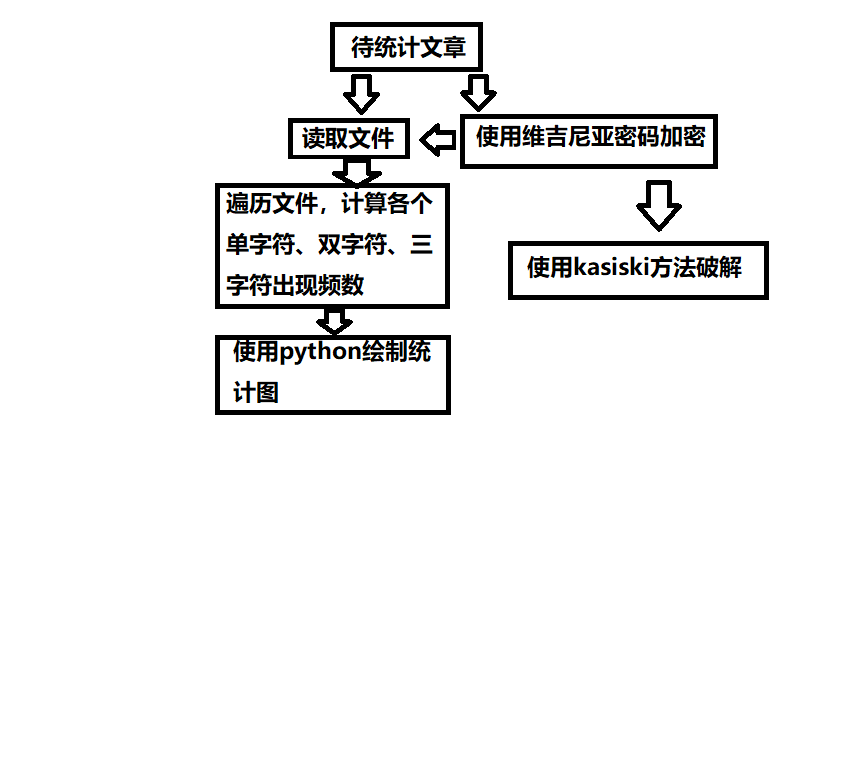
需求分析：读取一个文件并统计单字符、双字符、三字符的出现频率，分别用你直方图、三维直方图、三维密度图表示出来。字符范围应可以指定；画图部分可以包含在的软件之内，也可以单独使用某个工具实现。

本实验的所有代码都已经放在了附件里

## 算法分析

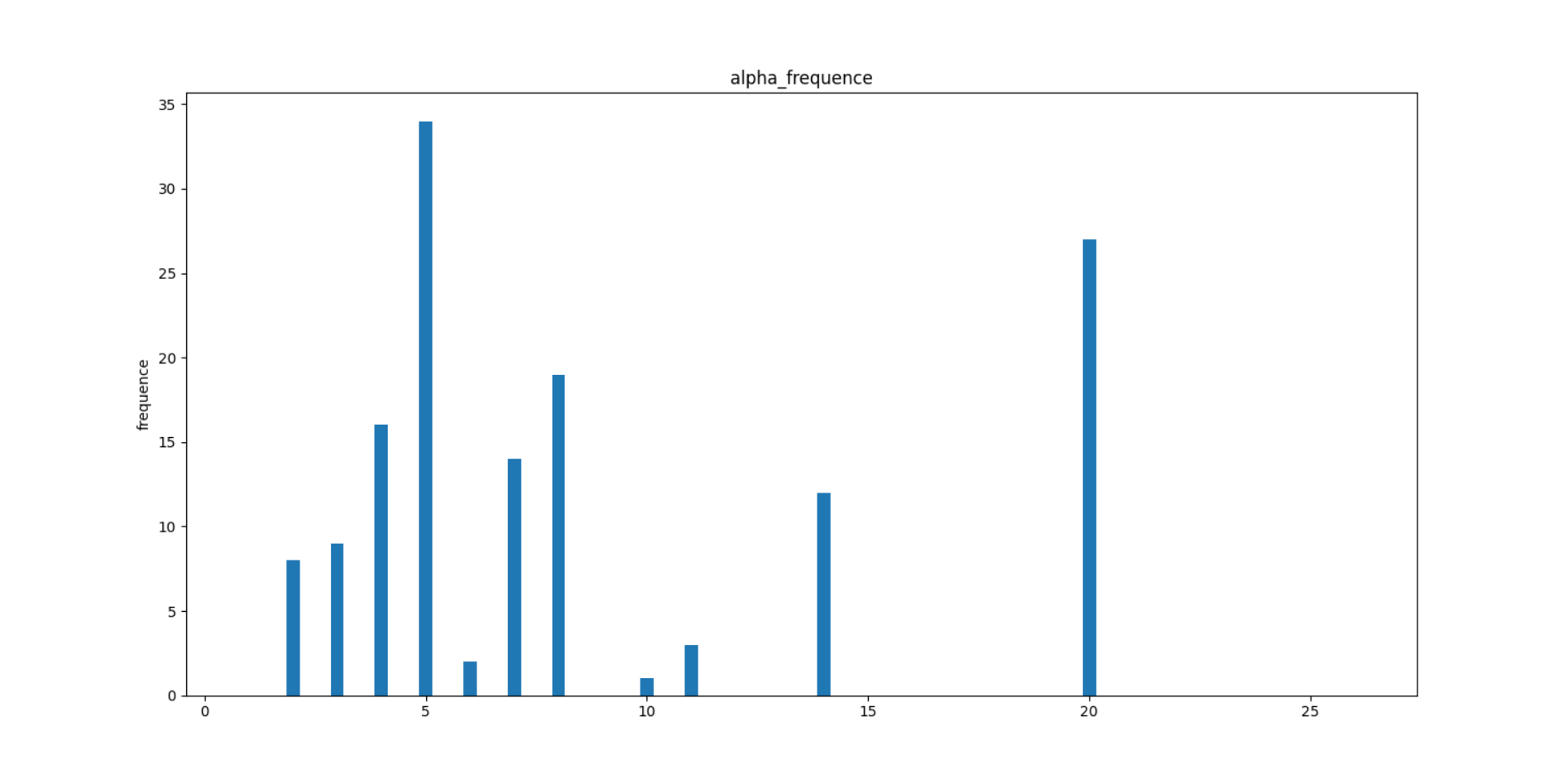
大致思路：统计部分使用c语言编写，先使用fileload函数读取文件内容存放在cipher数组里，再对cipher里的内容经行遍历，使用一维二维三维数组统计词频，每遇到一个字母或者双字符将对应位置的统计数组值加一，最后将文件保存在alpha\_frequence.txt中，（实现以上功能的代码都已经封装成函数并保存在frequence.h库文件中）

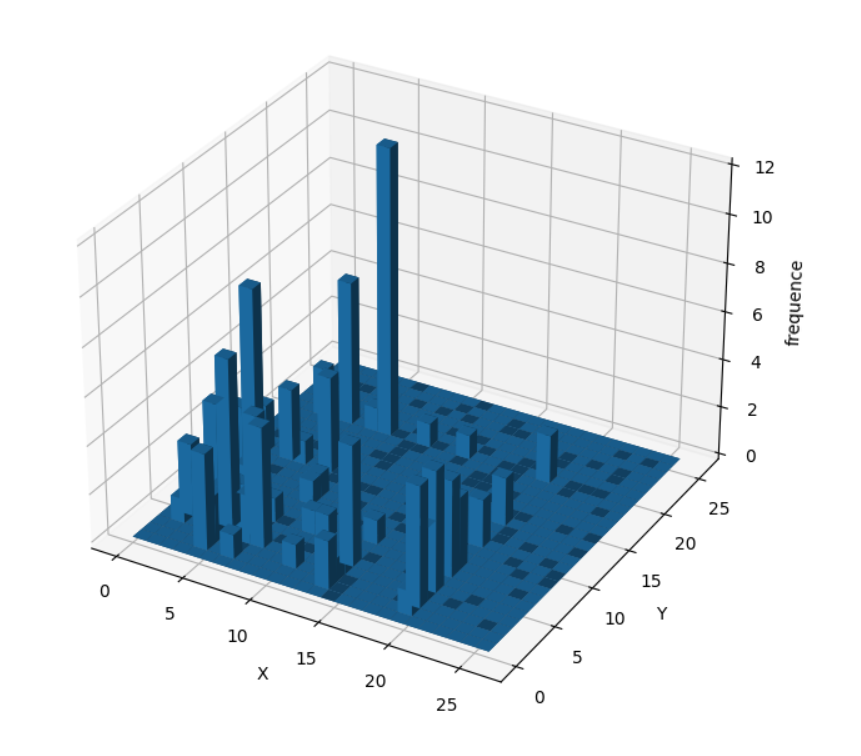
再执行system("python alpha\_frequence.py");来使用python绘制图表，使用了matplotlib库来进行表格的绘制。以实现对数据的可视化处理。

流程图如下：

## 完成效果

将要统计的文件命名为plaintext.txt文件里，执行程序后，会统计文件里的字母频数、双字母频数，并绘制出如下表格

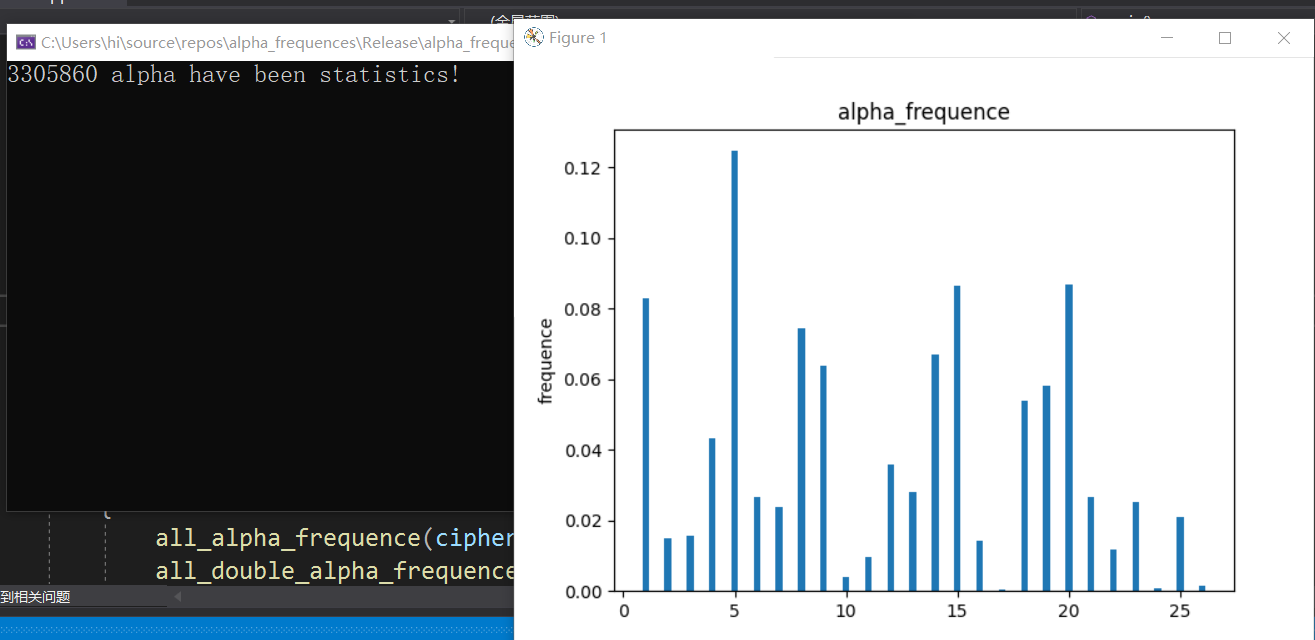




## 分析《圣经》中字母统计分布

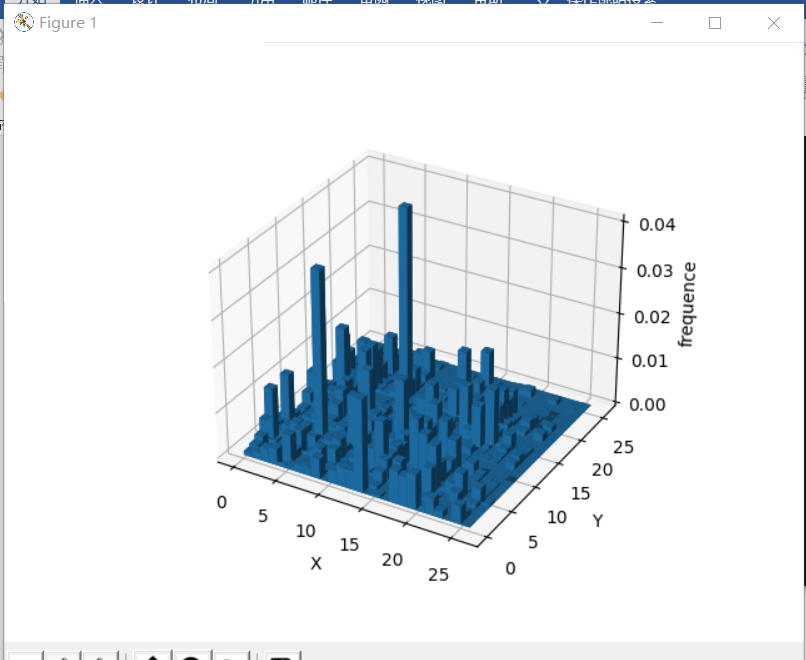
（字母改为小写，删除所有非字母的符号）

这里选用一本很长的的书《圣经》，一共3046103个字符，统计字母频率如下：

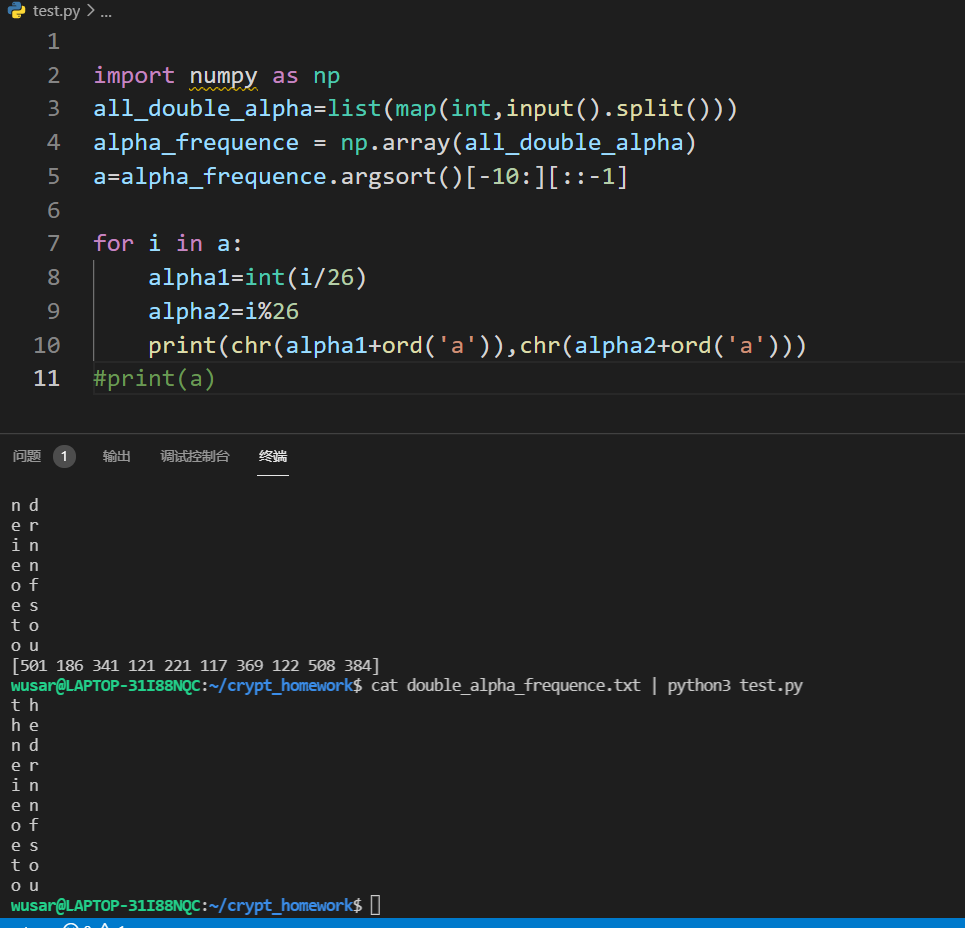


可以看出，字母e频率最高，其次是t、o、a，与英文字母出现频率相符合。

双字母频率如下：

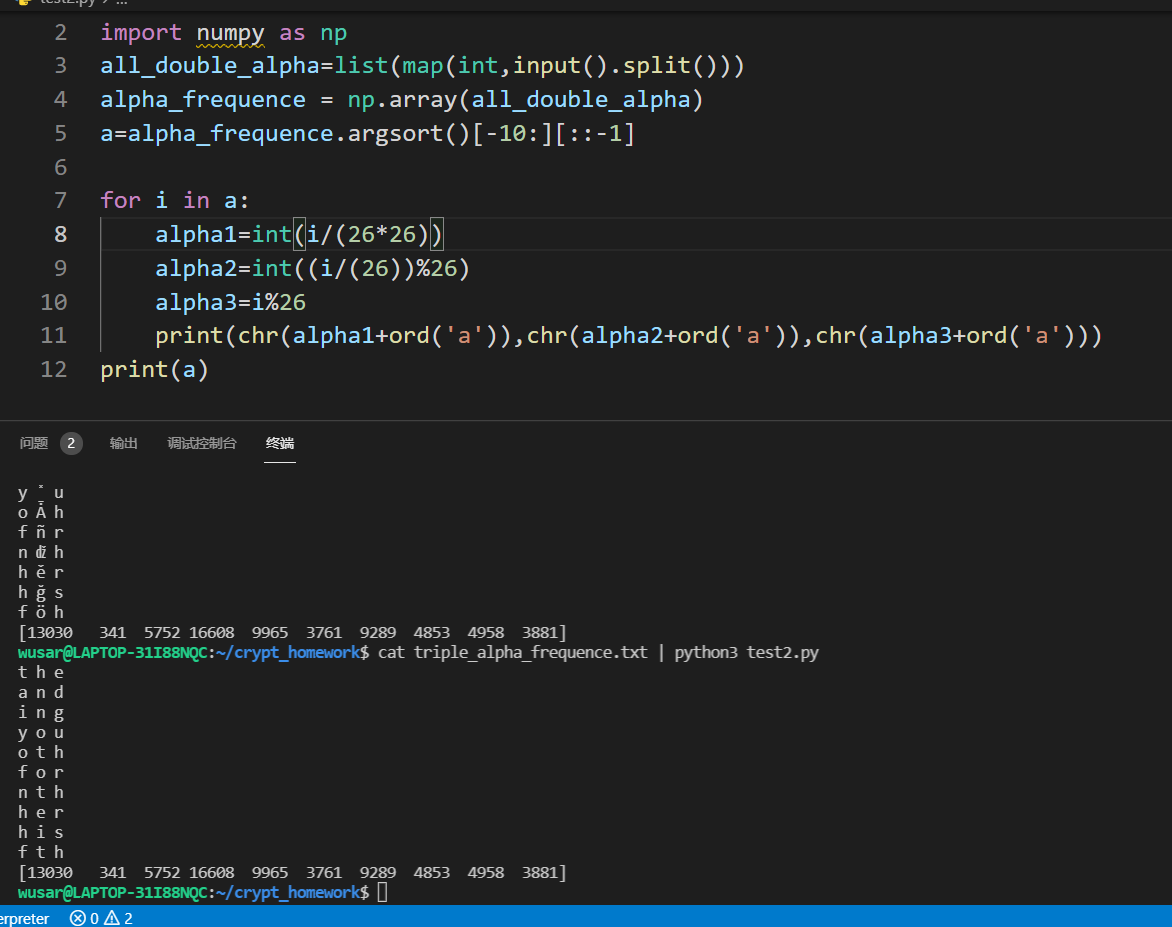


我们可以顺便分析一下文件double\_alpha\_frequence.txt，看一下出现频率前几的双字母：



出现频率前几的双字母有：th、er、in等，这些都是英文中常见的双字符，也是最常见的英文词根。

我们采用同样的方法分析一下三字母出现的频率：



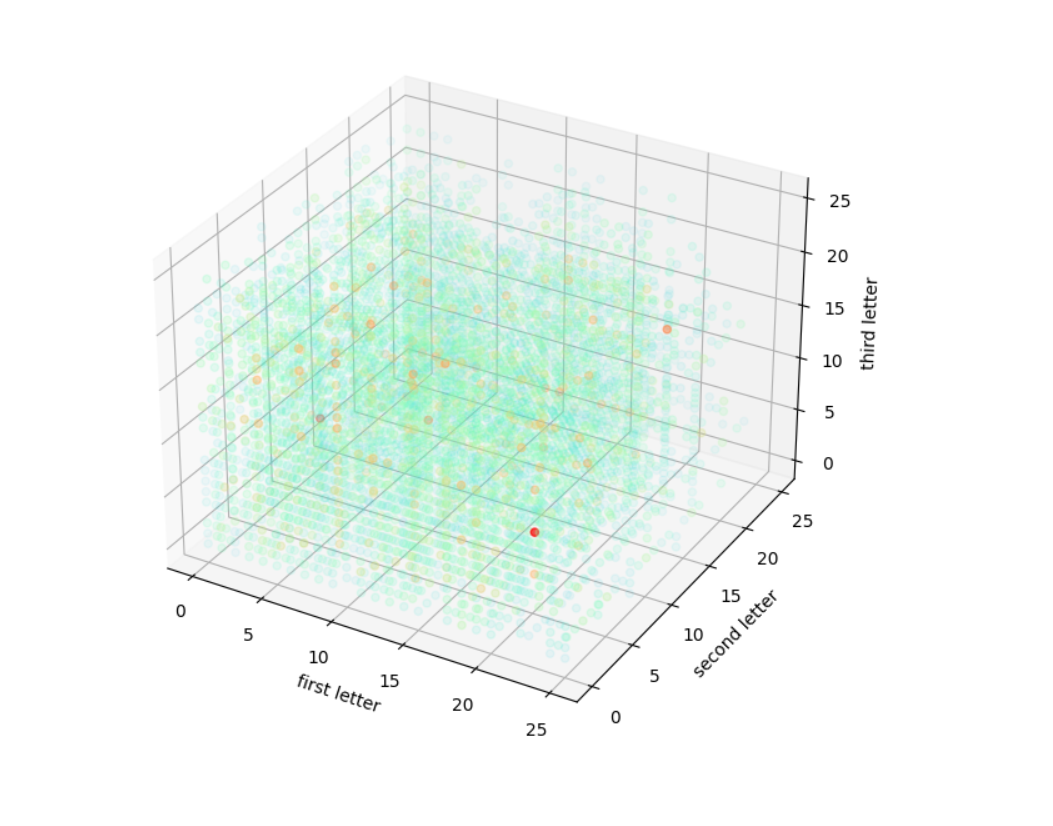
可以看到，出现频率前几的三字符为：the and ing等

### 三字符的可视化处理

至于三字符的可视化处理，就要麻烦很多

我们采用一个三维图中的一个点表示一个三字符的统计信息。

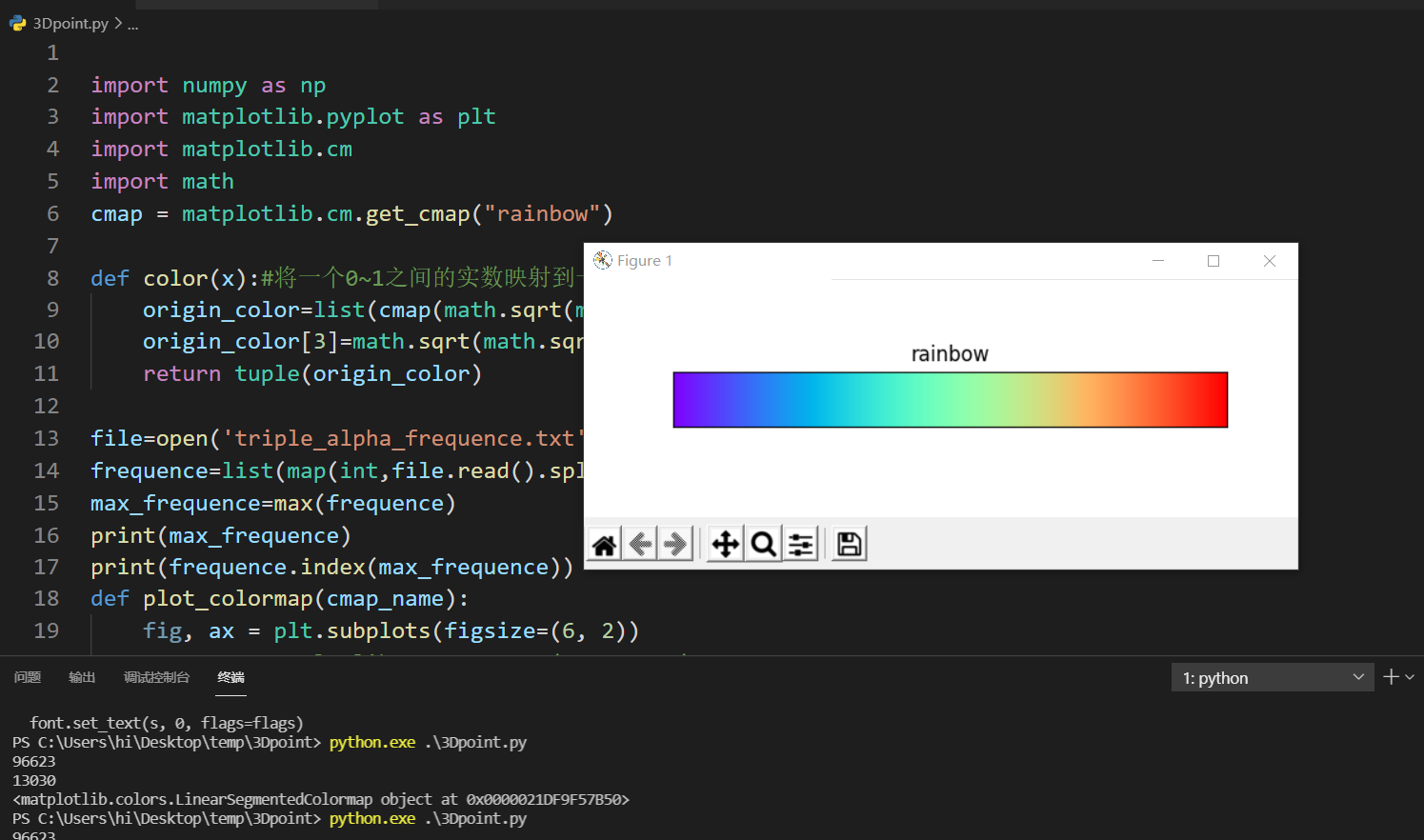
点的x，y，z三个坐标轴表示第1、2、3个字母，使用点的颜色和透明度代表对应三字符的出现频率。实现效果如下：



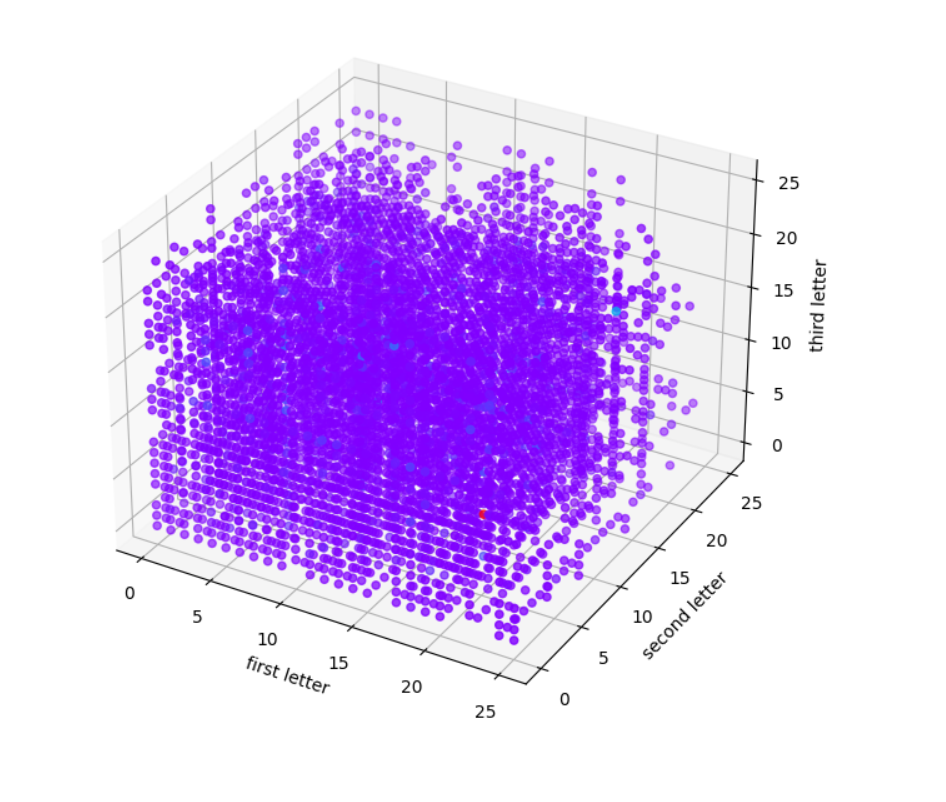
这里面最困难的部分在于从频率到颜色的映射，我使用matplotlib里面的colormap来实现。

Colormap将一个0~1之间的实数映射到一个rgb颜色三元组，如图：

最左边的蓝色表示0，最右边的红色表示1，这样就实现了颜色的表示



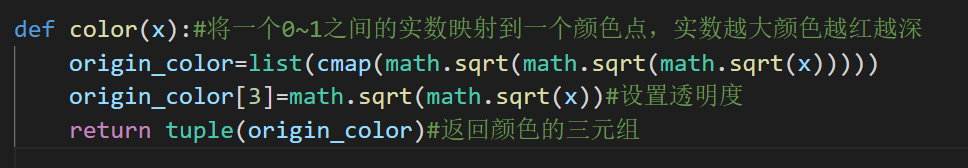
但是如果直接使用颜色表示，不经任何其他处理的话，图像就是这样：



可视化效果很差！

这是由于实际的三字符分布方差很大，出现频数高的三字符如“the”出现了96623次之多！而绝大部分三字符的出现频率都是0。

因此我们需要对颜色经行一些修正，修正的方法采用老师们经常使用的调分大法，直接对频率进行多次开方处理，并根据开方后的大小设置颜色，具体代码如下：

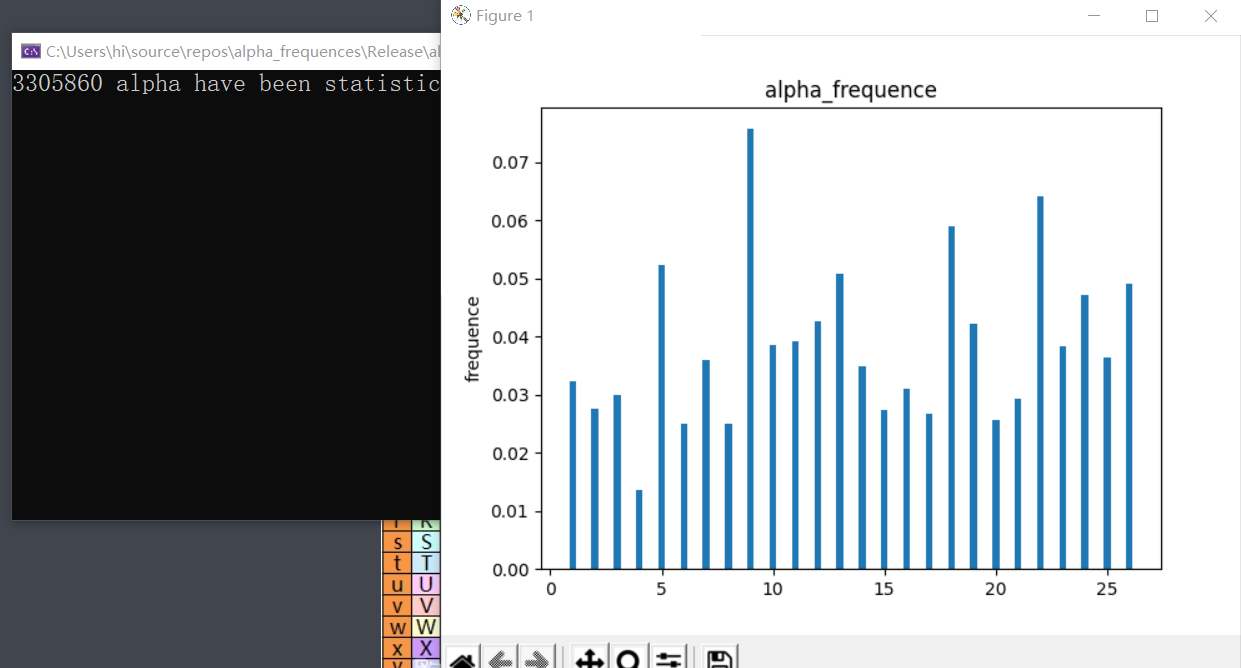


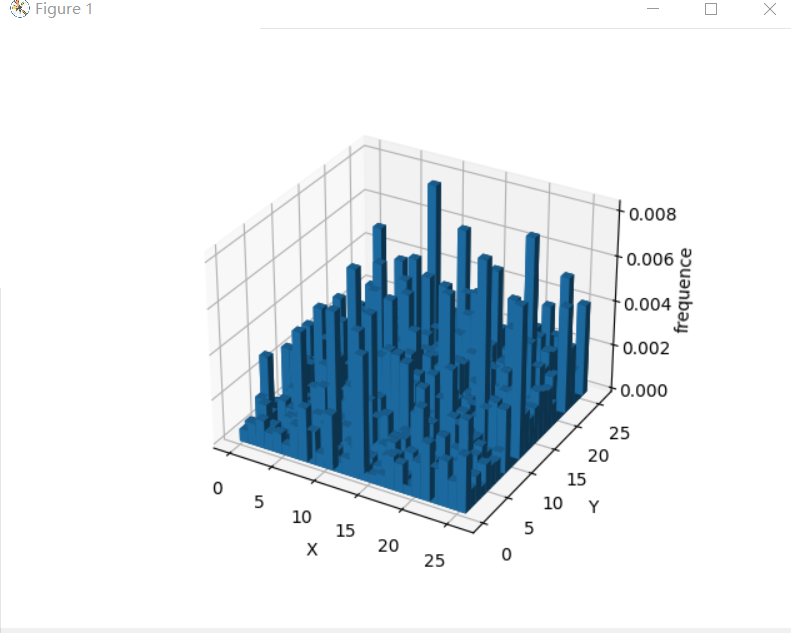
经测试，这样的调整能较好看的展现出三字母的频率分布。

## 用维吉尼亚密码加密这本书

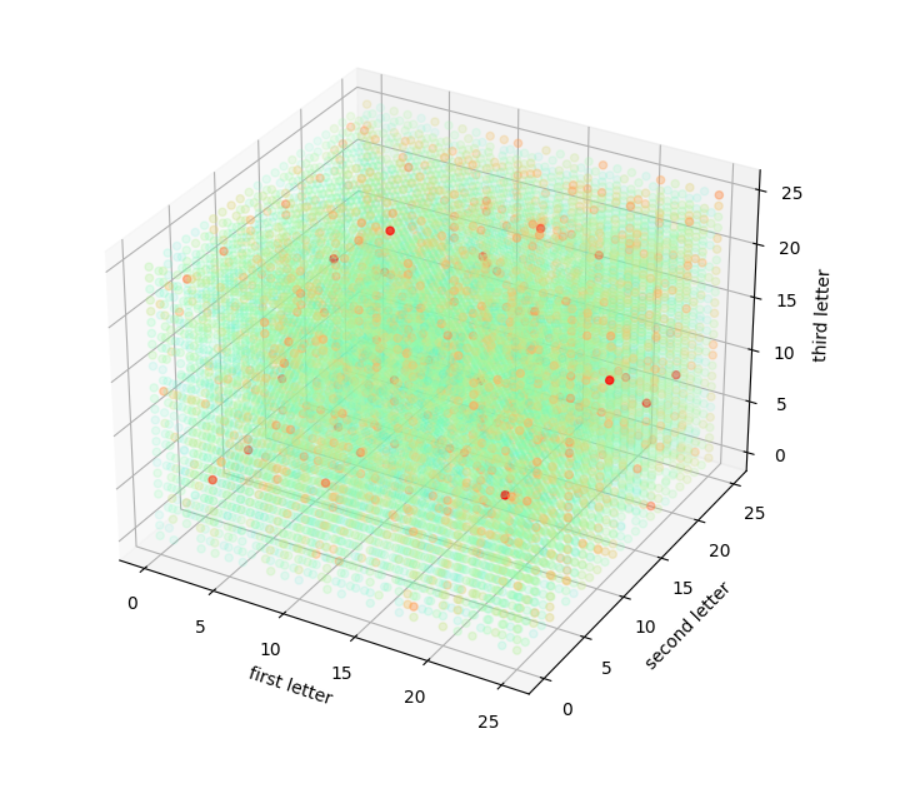
加密的代码写在vigenere.cpp中，读取plaintext.txt文件，根据key使用维吉尼亚密码加密该文件，将加密后的结果保存在cipher.txt中，再写一个vigenere\_decode.cpp代码，将加密后的cipher.txt 文件进行解密，将解密后的文件保存在decode.txt中，验证一下，可以看到decode.txt与plaintext.txt中字母相同。说明加密解密算法写的没有问题。

使用上面的字母频率统计工具，统计cipher.txt里的字母频率、双字母频率，如下：





三字符的统计分布图：



在这里，我发现使用维吉尼亚密码加密后的密文字母不是平均分布的！

在单字母频率分析中，密文里面字母i出现的频率最大，其原因也很好理解，在明文中字母e出现频率最大，密钥选用“vigenere”，e出现的次数也最多，而当明文为e，密钥为e时，加密出来的字母就是i！所以可以看出来，维吉尼亚密码并不能消除明文中的频率统计特征。

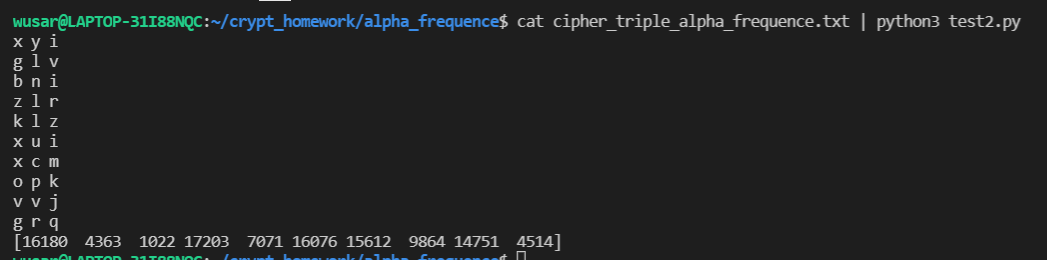
但是我们也看出，加密后密文的分布方差要明显比明文小很多，我们可以从信息熵的角度考虑这个问题：

H(M)=H(C|K)<=H(C)

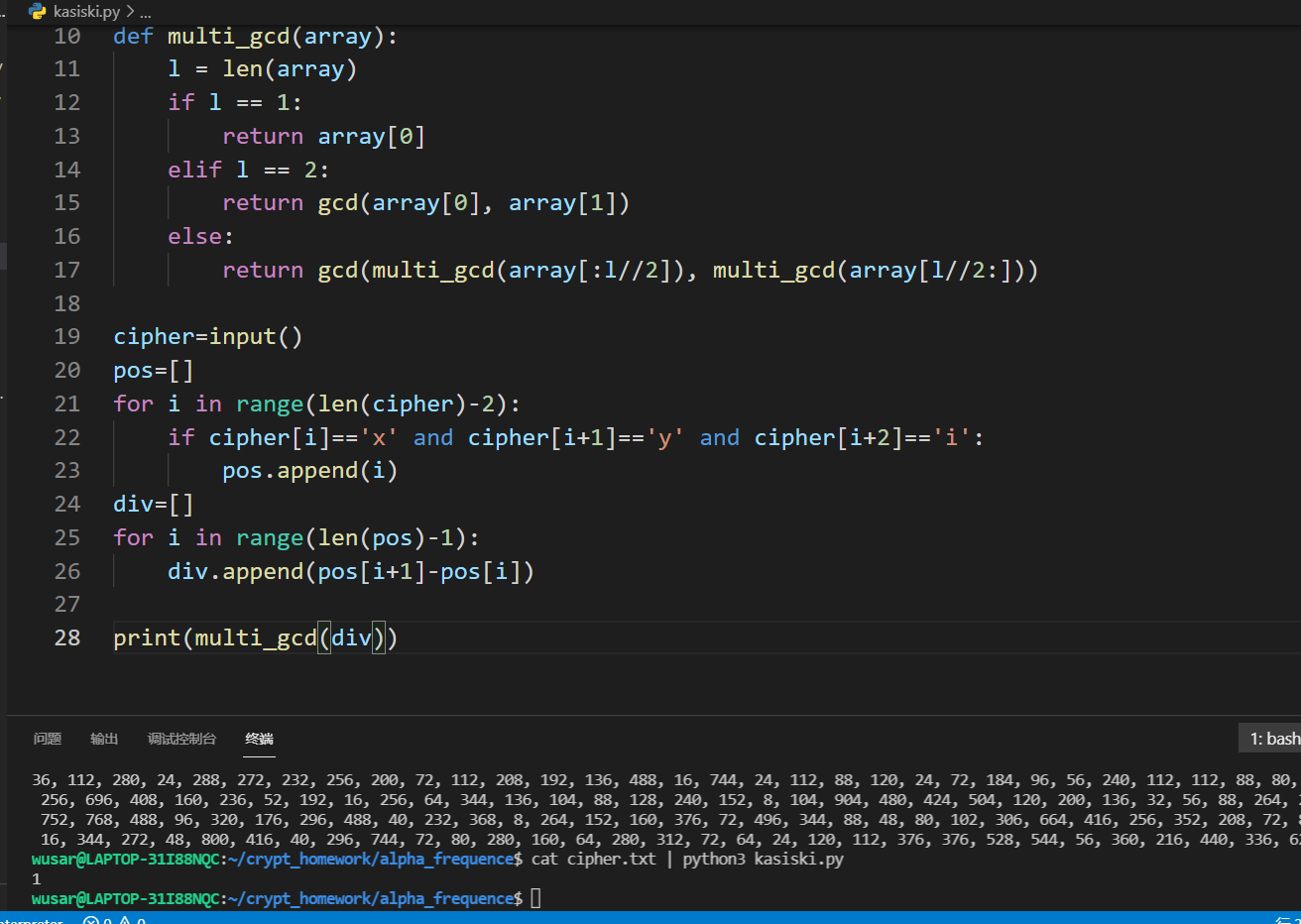
M、C、K为明文、密文、密钥，密文的熵是要小于明文的熵的。所以比起明文，密文会显得更加随机。

### 使用kasiski方法破解vigenere密码

这里我们选取密文中出现频率最大的三字符进行分析

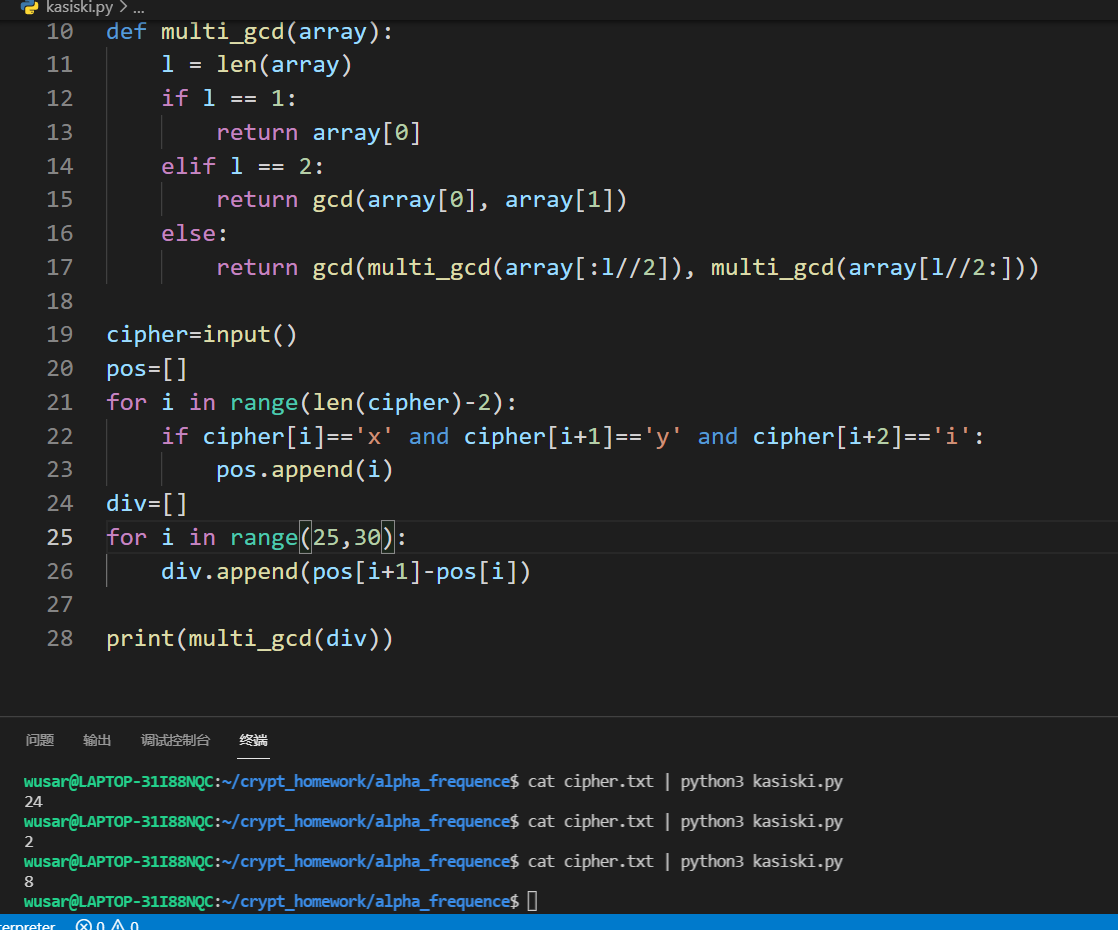


但是当我们直接求这个三字符的出现位置的差的最大公因数的时候，就会发现求出来的结果是1。



这是因为这个三字符在密文中出现的次数太多了，所以不满足kasiski方法的使用条件：不同的明文获得相同密文的巧合很少发生。

所以我们可以只选择较少的一部分密文进行分析，如图：



这样就可以求出来密钥长度应该是8

# 参考文献

无