

# C++基础与深度解析

## 第三次作业

### 项目 1：简单文件加密解密系统

这个项目的目标是构造一个简单的文件加解密系统。

我们将采用一种非常简单的策略实现文件的加解密。总体来说，我们需要构造两个程序 encrypt 与 decrypt，其使用方式为：

encrypt 码本文件名 输入文件名 输出文件名

decrypt 码本文件名 输入文件名 输出文件名

为了实现加解密，我们首先需要有一个码本文件。一个基本的码本文件包含 26 个字母，a~z 打乱顺序的结果，比如：qwertyuiopasdfghjklzxcvbnm。码本文件的第一个字母表示 a 应当替换成的字符（该案例中，a 应替换成 q），第二个字母表示 b 应当替换成的字符（该案例中，b 应替换成 w），依此类推。

在获取了这个码本文件后，encrypt 需要对输入文件进行加密，即将相应的字母进行替换。比如，如果输入文件中包含了如下的内容：

hello world

基于前文所述的码本文件，系统应当输出：

itssg vgksr

注意：在替换时，会有一些不属于 a~z 的字符，比如输入中的空格，这些字符应原样输出。

相应的，decrypt 应当接收码本文件与加密后的文件作为输入，以解密后的文件作为输出。

### 扩展 1：

现在，我们将整个系统进行扩展一下：码本文件将包含 256 个数字，为 0~255 打乱顺序的结果，比如：

123 234 1 0 255...

其中的第一个数 123 表示字符(char)0 应当被替换为(char)123，第二个字符表示(char)1 应当被替换为(char)234，依此类推。

基于这个码本文件，修改 encrypt 与 decrypt，使得它可以为任意文件进行加密解密。

注意：如果要支持任意文件的加密解密，那么就需要使用二进制的方式打开文件。

### 扩展 2：

尝试加密解密一个可执行文件（比如你的 C++ 开发环境中包含的可执行文件），看它是否可以正常工作。注意：在尝试之前，一定要做好文件备份，避免因你的程序出错而导致原始的可执行文件无法运行。