

汉字花样显示

郭泽均 3170105271

编译方法

在macOS或者Linux环境下，在 `code` 文件夹下，直接执行 `make` 指令编译源代码，输入指令 `./cndp` 执行程序。

在Windows环境下，手动编译执行源代码文件 `chinese_character_display.cpp`，之后执行可执行文件。

实现细节

课程提供的汉字库是遵循是 GB2312 编码所以在建立查找的列表时使用的文件必须使用 GB2312 编码或者互相兼容的编码，以免在相应的字库中找不到对应的汉字，以至于输出的结果与我们想要的不对应。

代码首先将 `pingying` 文件读入内存中，建立一个 `map` 类型是 `map<string, string>` 第一个属性对应的是汉字的拼音，有利于我们在设计输入法的时候查找汉字。第二个属性对应的是汉字，第一个字节对应的是 GB2312 编码中汉字的区，第二个字节对应的汉字的位。通过这两个属性，我们就可以在字库中查找到我们需要的汉字。将汉字的点阵信息读入到二维数组 `char mat[16][2]` 中，这样设计点阵信息的目的是要方便在镜像和反向显示汉字的时候方便写代码。

在显示汉字的时候使用一个函数，我们只需要给它不同的输入就可以得到不同的显示效果，这个函数具有很好的重用性。支持任意特殊效果的组合。并且支持不同程度的缩放和倾斜。这个函数实现的功能有：倾斜，缩放，反向，镜像。这四个效果可以进行任意的组合。

获取汉字偏移量的方法

- 偏移区块数 `qh`：第一个字节的值减去 `0xA1`，因为汉字区块区域是从 `0xA1` 区开始的。
- 偏移位码数 `wh`：第二个字节的值减去 `0xA1`

`offset = (94 * qh - wh) * 32`，按照这个偏移量从 `Hzk` 文件中读取对应的点阵信息，存放在一个二维数组中。遍历这个数组，检查每一个位置的值，使用判别式 `mat[j][i]&(0x80>>k)` 如果这个值是真，那么这个位置应该输出一个真值，反之输出一个假值。最后输出的结果就是汉字的点阵。