

实验报告

MF1933058 刘凡维

实验环境：

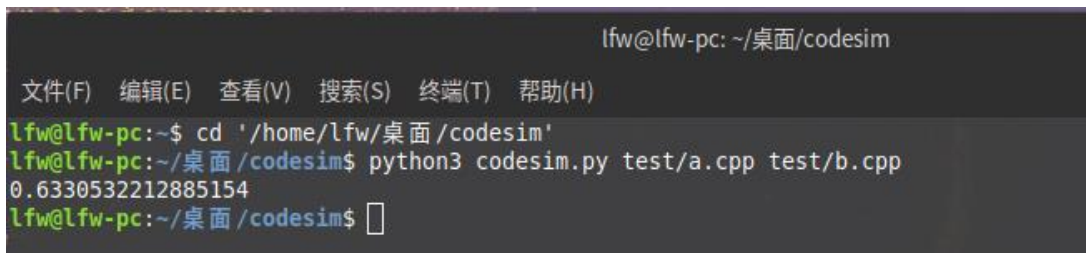
Ubuntu 18.4.1 Python 3.6.8

执行方法：

首先需要确保 python3 已经安装了 pygments 模块，如果缺少该模块，程序无法执行，需通过命令：`sudo pip3 install pygments` 安装该模块

执行文件 `codesim.py`：在文件根目录下使用命令：`python3 codesim.py file1 file2` 其中 `file1` 与 `file2` 为需要比较的两个 `cpp` 文件。若两个文件与 `codesim.py` 不在相同目录下，参数 `file1` 与参数 `file2` 需要包含文件完整路径。

比较 `test` 文件夹中 `a.cpp` 与 `b.cpp` 代码相似度的运行结果结果截图：



```
lfw@lfw-pc: ~/桌面/codesim
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
lfw@lfw-pc:~$ cd '/home/lfw/桌面/codesim'
lfw@lfw-pc:~/桌面/codesim$ python3 codesim.py test/a.cpp test/b.cpp
0.6330532212885154
lfw@lfw-pc:~/桌面/codesim$
```

算法思想：

首先分别读入两个 `cpp` 文件，对文件进行预处理，过滤掉头文件、注释、括号、分号等 `tokens`。此外，将具体变量名、函数名、字符串也转化为统一表示。将处理后的文件的每一个 `token` 读入并连接成一个新的字符串。于是将代码相似度度量转化为字符串相似度度量的问题。

在 Linux 下比较代码最简单的方式就是可以使用 `diff` 命令，Python 标准库下的 `difflib` 模块与 Linux 的 `diff` 命令相似，因此考虑可以使用该模块对比代码间的差异。`difflib` 模块下的 `SequenceMatcher` 类中的 `ratio()` 函数可以返回一个度量

两个字符串相似度的值，值在[0,1]之间。本次实验也使用了这个函数来度量两个c++代码间的相似度。

此外，通过调用 Python 中的 `pygments` 模块，对现有代码稍加改动便可以实现定位疑似抄袭代码的位置的功能，`pygment` 可通过对相似代码的高亮显示来使疑似抄袭代码可视化。

在读入包含中文注释的 `cpp` 文件时，在 `Pycharm` 中可能会存在中文显示为乱码的问题，最终导致 `codsim.py` 文件运行报错。这时可以考虑将 `Pycharm` 右下角的 `File Encoding` 设置为 `GBK` 格式然后在出现的提示框中选择“`Reload`”。然后在将 `File Encoding` 重新设置为 `UTF-8` 格式，此时在出现的提示框中选择“`Convert`”选项。最后重新运行程序，错误得到解决。