

CSE Lab3

By 彭超 黄子豪 沈云飞 俞哲轩

Description

本次 Lab 中，你将需要查阅资料，针对 12 月课程中所涉及到内容，完成一组练习。

Submission

- **Deadline: 2021.12.26, 23:59**
- 提交方式：请将说明文档（内容详见下文）和 Part B 代码文件打包为 `学号-姓名-lab3.zip` 上传到 elearning 的 lab3 提交栏目。
- **注意：抄袭零分**

Part A

- 请根据以下提示完成说明文档的 Part A：
 1. **在你的说明文档中**实现一个简单的汉明距离计算函数。以 Java 为例，代码模板如下：

```
1 public int hammingDistance(int x, int y) {
2     // TODO
3 }
```
 2. 请结合实例谈谈你是如何认识计算机系统中的可靠性的，简要（200字以内）说明在这一实例中可靠性是如何得以实现的。你可以以自己的实践经验为依据，也可以查阅资料。

Part B

- 使用代码实现修改一个文本文件（将其视为一个“数据库”）的程序，要求实现如下功能：
 - 文本文件的原始内容有 10 行，每行只有一个 "0" 字符和回车符。
 - 修改文件内容时需要调用 `update()` 把各行的数据均修改为某一新字符，比如 "1" 或 "2" 等字符。每次只能先读取某一行的字符、修改为新字符，再 `sleep()` 1 秒后方可继续修改文件，直到修改完毕。
 - 考虑到单个事务的 All-or-nothing 问题，如果进程运行完毕则所有行会被完全修改，而如果在进程执行过程中中断进程，文件会处于数据不一致的状态。因此，需要实现一个 Write-ahead log；在下一次重新运行程序之前，还需要让文本文件中所有行的数据保持一致。
 - 考虑到多个事务的 Before-or-after 问题，需要使用 Read-capture 策略。
- 请参考课程内容，**使用 Java 语言**完成。以下 API 可供参考：

```
1 /**
2  * You can alter these APIs if you wish.
3  * If you do so, please specify in your docs what you changed and why.
4  */
5 public class MyAtomicity {
6     // TODO: declare your variables here
7
8     /**
9      * Update the text file "database"
10     * @param ch The input value for all lines
```

```

11     */
12     public void update(char ch) {
13         // TODO
14     }
15
16     /**
17      * Write a single character to "database"
18      * @param index The location of database to be updated
19      * @param ch The input value
20      */
21     private void write(int index, char ch) {
22         // TODO
23     }
24
25     /**
26      * Log some information
27      * @param text Log text
28      */
29     private void log(String text) {
30         // TODO
31     }
32
33     /**
34      * Recover the system from your log.
35      * This should be called at the start of each {@code update()} call.
36      */
37     private void recover() {
38         // TODO
39     }
40 }

```

- 在实现过程中，请考虑以下需求：
 - 面向对象，即考虑模块化与封装性等；
 - 编码风格。
- 说明文档：
 - 请指出你是如何测试你的代码的，并附上测试截图；
 - 请简要说明你的设计决策。
- 附加内容，**请在说明文档中注明**你实现的项目：
 - 可尝试实现一组自动化测试用例；
 - 可尝试实现一个直观的用户界面，展示你的设计决策如何影响程序运行；
 - 其他你认为可以实现的附加内容。

Grading

得分项目	对应分数
Part A	30
1. 汉明距离	10
2. 可靠性	20
Part B	70
1. 功能实现与设计决策	40
2. 说明文档与测试结果	20
3. 面向对象与代码风格	10
附加内容	10
满分	100

- 在满分为 100 分的前提下，Part B 附加内容可占总分的 10%。