Operating System Homework 2

繳交期限: 4/8 17:00 前繳至 ED817

手寫題：

2.5 Why is the separation of mechanism and policy desirable?

2.6 Would it be possible for the user to develop a new command interpreter using the system-call interface provided by the operating system?

2.8 What is the main advantage for an operating-system designer of using a virtual-machine architecture? What is the main advantage for a user?

2.10 What is the main advantage of the layered approach to system design? What are the disadvantages of using the layered approach?

2.12 Describe three general methods for passing parameters to the operating system.

2.13 What is the main advantage of the microkernel approach to system design? How do user programs and system services interact in a microkernel architecture? What are the disadvantages of using the microkernel approach?

2.15 What are the two models of interprocess communication? What are the strengths and weaknesses of the two approaches?

2.18 How could a system be designed to allow a choice of operating systems from which to boot? What would the bootstrap program need to do?

2.21 Why do some systems store the operating system in firmware, while others store it on disk?

程式題：

2.22 In Section 2.3, we described a program that copies the contents of one file to a destination file. This program works by first prompting the user for the name of the source and destination files. Write this program using either the Win32 or POSIX API. Be sure to include all necessary error checking, including ensuring that the source file exists.

Once you have correctly designed and tested the program, if you used a system that supports it, run the program using a utility that traces system calls. Linux systems provide the `ptrace` utility, and Solaris systems use the `truss` or `dtrace` command. On Mac OS X, the `ktrace` facility provides similar functionality. As Windows systems do not provide such features, you will have to trace through the Win32 version of this program using a debugger.

Description:

本題由於目前課程才剛提到 child、parent process 的概念，因此同學僅需在附件 code 中提示的部分完成 file copy 的程式即可；但仍需照上方敘述所述，必須讓使用者輸入 source file、destination file name，並且實作需要的 error checking，剩餘的部分 ptrace 會追蹤同學 file copy code 中有使用的 system call 並印出，且使用哪些 system call 會做為評分依據。

Input:

1. Source file name: 長度不會超過 20 個字元
2. Destination file name: 長度不會超過 20 個字元

Output:

1. System call 0 (read): 2 分
2. System call 1 (write): 2 分
3. System call 2 (open): 2 分
4. System call 3 (close): 2 分
5. System call 5 (fstat): 2 分

Testing Example:

```
(gongenv) root@s040130:~/gong/OS/HW2# gcc -std=c11 -o hw2_22_309511042 hw2_22_309511042.c
(gongenv) root@s040130:~/gong/OS/HW2# ./hw2_22_309511042
system call: 62
system call: 5
system call: 5
system call: 12
system call: 12
system call: 12
system call: 12
system call: 1
Please input the source file name:
system call: 1
system call: 5
system call: 5
system call: 0
input.2.txt
system call: 0
system call: 2
system call: 2
system call: 1
Failed to open file input.2.txt
system call: 1
system call: 1
Please input the source file name:
system call: 1
system call: 0
input.txt
system call: 0
system call: 2
system call: 2
system call: 1
Please input the destination file name:
system call: 1
system call: 0
output.txt
system call: 0
system call: 2
system call: 2
system call: 5
system call: 5
system call: 0
system call: 0
system call: 5
system call: 5
system call: 0
system call: 0
system call: 1
output.txt has generated
system call: 1
system call: 3
system call: 3
system call: 1
system call: 1
system call: 3
system call: 3
system call: 231
```

Rules:

1. 請確保你的.c/.cpp 檔能被 gcc/g++編譯 (不能的話則扣 5 分)
2. 請確保你的程式能 run 在 linux 的環境上 (不能的話則扣 5 分)
3. 請將上傳至 E3 的檔案命名為 hw2_22_學號.c/.cpp (不符合規定扣 2 分)

Reference:

1. Linux system call table: https://chromium.googlesource.com/chromiumos/docs/+/master/constants/syscalls.md
2. Linux man page: https://man7.org/linux/man-pages/