**系統程式期末考（筆試部分）**

|  |  |  |
| --- | --- | --- |
| **姓名：** | **學號：** | **成績：** |

**計分方式：每題7分，最高分為100分。**

1. 請簡單比較一下傳統Linux的權限控制及ACL（例如：使用getfacl和setfacl）的權限控制（舉一個例子說明傳統Linux的權限控制做不到，而ACL做得到）
2. 請問如何用google查詢在Windows底下，監測檔案系統的函數名稱（hint：用Linux的關鍵字搭配作業系統名稱）
3. 目前在Linux系統中只有「一個」行程（該行程為prog）是活躍的狀態（active，換句話說就是需要CPU），其他行程都是inactive的狀態（目前不需要CPU）。請問下列這些指令會讓這個行程的執行時間造成什麼樣的影響（hint: nice -n -5是提高此行程對其他行程的優先權，nice -n 5是降低此行程對其他行程的優先權）

sudo nice -n -5 prog

sudo nice -n 5 prog

1. 請問下列程式的執行結果，請解釋你的答案

printf(“hello ”);

fork();

printf(“Ron\n”);

1. 請描述如何同時對fork進行除錯（必須同時對parent和child除錯）（hint：attach，set waiting = 0，while loop）
2. 請問何謂殭屍行程（提示：fork、wait）
3. system()和execv()都可以執行外部程式（例如：ls），請問下列程式碼各會在螢幕上印出多少個“fork”

|  |
| --- |
| 1. /\*程式一\*/ 2. void main(int argc, char\*\* argv) { 3. system(“ls”); 4. fork() 5. fork(); 6. printf(“fork\n”); 7. } |
| 1. /\*程式二\*/ 2. void main(int argc, char\*\* argv) { 3. char \*const paramList[] = {"/bin/ls", "-l", NULL}; 4. execv("/bin/ls", paramList); 5. fork(); 6. fork(); 7. printf("fork\n"); 8. } |

1. 請問如何使用signal函數攔截ctr-c這組按鍵輸入嗎？（hint：SIGINT）
2. Linux中signal中編號1-31，與編號SIGRTMIN~SIGRTMAX這二個族群有何不同？（提示：如果連續、大量的發送signal各會發生什麼樣的事情？）
3. 程式碼如下：程式碼如下圖所示。其函數呼叫為A()➜B()➜C()。請問當C()執行完回到B()的時候，可以在B中使用longjmp跳到C函數嗎？請解釋你的答案。（請參考提示，這一題答「可以或不可以」都可能是對的答案，重點在於你怎樣處理堆疊）

|  |
| --- |
| 1. sigjmp\_buf jmp\_buf\_a, jmp\_buf\_b; 2. A() { setjump(&jmp\_buf\_a); B(); } 3. B() { 4. setjmp(&jmp\_buf＿b); 5. C(); 6. longjmp(&jmp\_buf\_c, 1); /\*這一行可以嗎？\*/ 7. } 8. C() { setjmp(&jmp\_buf\_c);} |

第十題的提示

一張含有 螢幕擷取畫面 的圖片

自動產生的描述

1. 請解釋dup這個函數的用途，請說明下列程式碼中printf(“hello\n”)會將資料寫到哪個地方（pipe？螢幕？）

|  |
| --- |
| 1. **pipe(pipefd);** 2. ret = fork(); 3. if (ret==0) { /\*child\*/ 4. close(0); 5. **dup(pipefd[0]);** 6. **close(pipefd[1]); close(pipefd[0]);** 7. printf("hello\n"); 8. } else { /\*...\*/} |

1. 請問pipe可否「繼承」給子行程，如果沒有特殊的設定，請問執行execve-like的系統呼叫以後，pipe可否繼承給新載入的程式碼（提示：在Linux內可以打像下列這樣的指令：$ ls -R / | sort，bash如何達成這樣的指令呢？）
2. 請問在Linux中，如果父行程已經結束，那麼子行程會自動結束嗎？子行程將來會變成zombie嗎？（請從「會、可能會、一定不會」三個當中選出你的答案）
3. 請問在Linux中，那二個signal無法使用signal函數「抓住」
4. 請問在Linux中，如果使用read()讀取一個超大的檔案（例如：2GB） 到記憶體中。讀取的過程中剛好使用者按下ctr-c。假設使用者使用signal()幫ctr-c安裝了signal handler，其程式碼如下。請問當使用者按下ctr-c以後，該read的行為會是如何？

void ctr\_c(int signum) {

printf(“reading…\n”);

}