

Assignment7

Team18

112062519 廖思愷

112062636 游竣量

111065547 游述宇

1. Fork a new child.

創建子進程，如果 fork 失敗，打印錯誤信，pid == 0 代表子進程要執行的程式碼區段。

```
if ((pid = fork()) == -1) {  
    perror("fork error");  
    exit(1);  
} else if (pid == 0) {
```

2. Create a new session in the child process.

子進程創建新的 session，如果失敗則打印錯誤信息。

```
if (setsid() == -1) {  
    perror("session error");  
    exit(1);  
}
```

3. Verify that the child is a process group leader and no longer has a controlling terminal by using ps command to print PID, PGRP, and TPGID.

setsid() 前，獲取當前 TPGID，並印出子進程的 PID、進程組 ID (PGRP) 和 TPGID 信息。setsid() 後，再次獲取當前 TPGID，並印出子進程的 PID、進程組 ID (PGRP) 和 TPGID 信息，最後執行 ps 命令，查看子進程狀態。

```

} else if (pid == 0) {
    pid_t tpgid = tcgetpgrp(STDIN_FILENO);
    printf("Before setsid() Child PID: %d PGRP: %d TPGID: %d\n", getpid(),
        getpgrp(), tpgid);
    if (setsid() == -1) {
        perror("session error");
        exit(1);
    }
    tpgid = tcgetpgrp(STDIN_FILENO);
    printf("After setsid() Child PID: %d PGRP: %d TPGID: %d\n", getpid(),
        getpgrp(), tpgid);

    char cmd[50] = {0};
    snprintf(cmd, sizeof(cmd), "ps -o pid,pgid,tpgid,comm -p %d", getpid());
    system(cmd);
    return 1;
}

```

輸出：

```

./assignment7
Before setsid() Child PID: 51976 PGRP: 51975 TPGID: 51975
After setsid() Child PID: 51976 PGRP: 51976 TPGID: -1
  PID  PGID  TPGID  COMM
51976 51976      0 ./assignment7

```

4. Explain why the child process does not have a controlling terminal. What are PID, PGRP, and TPGID values in this case? What are their meanings? Please answer these questions in your report.

PID (Process ID)：進程的唯一標識。

PGRP (Process Group ID)：進程組的標識符。

TPGID (Terminal Process Group ID)：這是進程與終端關聯的前台進程組的 ID。

當子進程調用 `setsid()` 函數創建新的 session 時，子進程成為 new session 領導者，且進程組 ID (PGRP) 設置為其自己的進程 ID (PID)，而且子進程與原始的控制終端脫離關聯，因此沒有控制終端，所以 TPGID 為 -1。由結果能觀察到 `setsid()` 後，子進程的 STDIN 被改變 (-1，無)，但 STDOUT 仍然指向原終端（自動從父進程繼承），因此能正常輸出訊息。