Assignment7

- Code

```
1  #include <unistd.h>
2  #include <stdio.h>
3  #include <fcntl.h>
4  #include <stdlib.h>
5
6  int main(int argc, char *argv[]) {
7      pid_t pid = fork();
8      if(pid < 0) {
9          perror("fork error");
10         return -1;
11     } else if(pid == 0) {
12         // Child
13         setsid();
14         /* To verify that child process does not have a control terminal
15         if((open("/dev/tty", O_RDWR)) < 0) {
16             printf("The child has no controlling terminal.\n");
17         } else {
18             printf("The child has a controlling terminal\n");
19         }
20         */
21     } else {
22         // Parent
23         // printf("child = %d\n", pid);
24         system("ps -x -o pid,pgid,tpgid");
25     }
26     return 0;
27 }
28
```

- Line 7
  (1) Fork a new child.
- Line 13
  (2) Create a new session in the child process.
- Line 14 to Line 19
  Try to open the controlling terminal by the child and find that the child has no controlling terminal.
- Line 24
  (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. The following table is our result.

PID	PGID	TPGID
10090	10090	0
1487	1487	10089
10089	10089	10089
10091	10089	10089

We find that the PID of the child process is 10090, and its TPGID is 0, which means it has no controlling terminal.
(4)
Explain why the child process does not have a controlling terminal.
Ans: 因為 child 在 setsid() 之後把自己變成一個新的 session, 而這個 session 不像是原本的 terminal, 比較像是在背景執行的新 session. 所以它沒有 controlling terminal.
What are PID, PGRP, and TPGID values in this case?
Ans: 原本 child 的 PGRP 會被改成自己的 PID, TPGID 會變為 0.
What are their meanings?
Ans: 代表 child 脫離原本的 parant, 自己變成一個 session.