Written Assignment #2

Due Date: Oct. 22, 2007 Monday

Instruction: Answer the following questions. Read the “Assignment and Examination Policy” Section of the Syllabus before you submit your final copy.

(50-point) Question #2: Summarize the manual pages of the following 10 system calls (5-point each): (1) fork, (2) vfork, (3) wait, (4) waitpid, (5) __clone, (6) execvp, (7) execve, (8) pthread_create, (9) pthread_exit, and (10) pthread_join. (“man fork”, “man vfork”, “man 2 wait”, “man waitpid”, “man clone”, … or Internet search).

(50-point) Question #3: A process tree is a hierarchy of processes. In a process tree, each node represents a process. In the following figure that shows an example of process tree, the process 3203 is a child of the process 3202.
Consider the following programs. Give the process tree and a possible output produced by the following code.

```c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>

main()
{
    int i, j;
    int Num_of_children;
    pid_t pid[10];
    pid_t return_pid;

    Num_of_children = 0;
    fork();
    return_pid = fork();
    if(return_pid > 0) {
        pid[Num_of_children] = return_pid;
        Num_of_children += 1;
    }
    for(i=0; i<3; i++) {
        return_pid = fork();
        if(return_pid > 0) {
            pid[Num_of_children] = return_pid;
            Num_of_children += 1;
            for(j=0; j<Num_of_children; j++) waitpid(pid[j], NULL, 0);
            execvp("./sample.out", NULL);
            fork();
        } else {
            return_pid = fork();
            if(return_pid > 0) {
                pid[Num_of_children] = return_pid;
                Num_of_children += 1;
            }
            fprintf(stderr, "PID = %d\n", getpid());
            fprintf(stderr, "PPID = %d\n", getppid());
        }
    }
    for(j=0; j<Num_of_children; j++) waitpid(pid[j], NULL, 0);
}
```

<sample.c program>

```c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>

main()
{
    fprintf(stderr, "Sample~in %d\n", getpid());
}
```