

Lab : Process and Programs

제출 내용

- 코드 1개, 문서파일 1개 (각 실행결과 화면 캡처 + (2)(3) 실행결과 분석)

제출기한: ~12:40

1. (waitdemo1.c 수정)

- 명령행 인자로 자식 프로세스의 수 n을 입력받음
- 부모는 n개의 자식 프로세스를 생성
- 각 자식은 1~5초 사이의 무작위 시간 동안 sleep한 뒤, 종료 코드로 그 시간을 반환
- 부모는 모든 자식이 종료할 때까지 wait, 자식이 exit될 때마다 자식의 PID와 status 코드를 출력

```
$ ./ex1 2
자식 프로세스 1001: 3초 동안 sleep
자식 프로세스 1002: 1초 동안 sleep
부모: 자식 1002 종료, 종료코드 1
부모: 자식 1001 종료, 종료코드 3
모든 자식 종료 완료
```

```
$ ./ex1 3
자식 프로세스 12345: 4초 동안 sleep
자식 프로세스 12346: 2초 동안 sleep
자식 프로세스 12347: 3초 동안 sleep
부모: 자식 12346 종료, 종료코드 2
부모: 자식 12347 종료, 종료코드 3
부모: 자식 12345 종료, 종료코드 4
모든 자식 종료 완료
```

```
$
```

2~3 (연습문제 8.4, 8.5) 코드 작성, 실행 결과 분석

- 8.4, 8.5 각 실행 결과 분석 요약

8.4 *fork and file descriptors* Consider this code:

```
main()
{
    int  fd;
    int  pid;
    char msg1[] = "Testing 1 2 3 ..\n";
    char msg2[] = "Hello, hello\n";

    if ( (fd = creat("testfile", 0644)) == -1 )
        return 0;
    if ( write(fd, msg1, strlen(msg1)) == -1 )
        return 0;

    if ( (pid = fork()) == -1 )
        return 0;
    if ( write(fd, msg2, strlen(msg2)) == -1 )
        return 0;
    close(fd);
    return 1;
}
```

Test this program. After the call to `fork`, both processes have a file descriptor set to the same current position in the output file. How many messages appear in the file? What does the number of lines tell you about file descriptors and connections to files?

8.5 *fork and standard I/O* Consider this code:

```
#include <stdio.h>

main()
```

```
{
    FILE *fp;
    int  pid;
    char msg1[] = "Testing 1 2 3 ..\n";
    char msg2[] = "Hello, hello\n";
    if ( (fp = fopen("testfile2", "w")) == NULL )
        return 0;
    fprintf(fp, "%s", msg1);

    if ( (pid = fork()) == -1 )
        return 0;
    fprintf(fp, "%s", msg2);
    fclose(fp);
    return 1;
}
```

Test this program. How many messages appear in the file? Explain the results. Compare this program to the output of `forkdemo1.c` in the text.