

**[506489] 시스템프로그래밍****실습 #07 문제 및 보고서**

<b>이름</b>	곽영주
<b>학번</b>	20175105
<b>소속 학과/대학</b>	빅데이터
<b>분반</b>	01 (담당교수: 김태운)

### <주의사항>

- 개별 과제입니다. (팀으로 진행하는 과제가 아니며, 모든 학생이 보고서를 제출해야 함)
- **각각의 문제 바로 아래에 답을 작성 후 제출해 주세요.**
  - 소스코드/스크립트 등을 작성 한 경우, 해당 파일의 이름도 적어주세요.
- **SmartLEAD 제출 데드라인:**
  - **월요일 분반: 다음 다음 실습 시간 전날까지(일요일 까지) // 2 주간 진행**
  - **수요일 분반: 다음 다음 실습 시간 전날까지 (화요일 까지) // 2 주간 진행**
  - 데드라인을 지나서 제출하면 24 시간 단위로 20%씩 감점(5 일 경과 시 0 점)
  - 주말/휴일/학교행사 등 모든 날짜 카운트 함
  - 부정행위 적발 시, 원본(보여준 사람)과 복사본(베낀 사람) 모두 0 점 처리함
  - 예외 없음
- **SmartLEAD 에 아래의 파일을 제출해 주세요**
  - **보고서(PDF 파일로 변환 후 제출 권장하나, WORD 형식으로 제출도 가능)**
  - 보고서 파일명에 이름과 학번을 입력해 주세요.
  - **소스코드, 스크립트, Makefile 등을 작성해야 하는 경우, 모든 파일 제출 (zip 파일로 압축하여 하나의 파일로 제출)**

### <개요>

이번 과제는 프로세스를 생성하는 것과 관련된 내용입니다.

## &lt;실습 과제&gt;

## [Q 0] 요약 [배점: 10]

이번 과제에서 배운 내용 또는 과제 완성을 위해서 무엇을 했는지 2~3 문장으로 요약하세요.

답변: 이번 과제를 통해 fork, wait, pid 의 개념을 다시 한 번 잡을 수 있었습니다. 그리고

간단하게 sleep 함수를 통해 프로세스 생성되고 제거되는 것을 직접 봄으로써 프로세스의 대한 개념을 잡는데 큰 도움이 되었습니다.

## [Q 1] htop [배점: 10]

터미널에서 \$htop 명령을 입력하세요 (설치방법: \$sudo apt install htop). CPU 사용률을 기준으로 프로세스를 정렬하세요(F6 키를 누르고, PERCENT\_CPU 선택 후 엔터). 결과 화면을 본 문서에 첨부하세요. 캡처한 화면을 기준으로 다음 질문에 답하세요. \$htop 명령어를 제외하면, 가장 많은 CPU 자원을 사용하는 프로세스는 무엇인지 순서대로 3 개를 골라서 해당하는 Command 를 아래에 입력하세요.

답변 (캡처화면):

```

CPU: 3.3% Tasks: 110, 232 thr: 1 running
Mem: [|||||] [557M/978M] Load average: 0.14 0.61 0.42
Swp: [|||||] 75.7M/2.00G Uptime: 00:22:57

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
 316 yeongju  20    0 21188 4636 2320 S  2.6 0.5  0:01.00 htop
1050 yeongju  20    0 3643M 295M 94396 S  0.7 26.1 0:09.38 /usr/bin/gnome-shell
1631 yeongju  20    0 319M 7952 7512 S  0.7 0.8  0:00.02 /usr/libexec/gvfs-fs-volume-monitor
1750 yeongju  20    0 160M 3660 3196 S  0.7 0.4  0:00.01 /usr/bin/VBoxClient --clipboard
1772 yeongju  20    0 161M 1852 1616 S  0.0 0.2  0:00.65 /usr/bin/VBoxClient --draganddrop
1769 yeongju  20    0 161M 1852 1616 S  0.0 0.2  0:00.65 /usr/bin/VBoxClient --draganddrop
2340 yeongju  20    0 1996 5816 4348 S  0.0 0.6  0:00.37 sshd: yeongju@pts/1
534 messagebu 20    0 9856 5660 3688 S  0.0 0.6  0:01.99 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
783 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.96 /usr/lib/snapd/snapd
527 root      20    0 242M 7364 6596 S  0.0 0.7  0:00.12 /usr/lib/accounts-service/accounts-daemon
906 root      20    0 298M 2756 2436 S  0.0 0.3  0:00.16 /usr/sbin/VBoxService --pidfile /var/run/vboxadd-service.sh
902 root      20    0 298M 2756 2436 S  0.0 0.3  0:00.21 /usr/sbin/VBoxService --pidfile /var/run/vboxadd-service.sh
570 root      20    0 242M 7364 6596 S  0.0 0.7  0:00.06 /usr/lib/accounts-service/accounts-daemon
649 root      20    0 337M 13972 11848 S  0.0 1.4  0:00.02 /usr/sbin/NetworkManager --no-daemon
1 root       20    0 163M 9548 7032 S  0.0 1.0  0:02.29 /sbin/init splash
213 root      19   -1 53348 16976 15784 S  0.0 1.7  0:00.72 /lib/systemd/systemd-journald
272 root      20    0 24200 4572 2728 S  0.0 0.5  0:00.96 /lib/systemd/systemd-udevd
491 systemd-t 20    0 23904 10040 7316 S  0.0 1.0  0:00.20 /lib/systemd/systemd-resolved
499 systemd-t 20    0 90260 5224 4604 S  0.0 0.5  0:00.00 /lib/systemd/systemd-timesyncd
492 systemd-t 20    0 90260 5224 4604 S  0.0 0.5  0:00.15 /lib/systemd/systemd-timesyncd
656 root      20    0 242M 7364 6596 S  0.0 0.7  0:00.01 /usr/lib/accounts-service/accounts-daemon
528 root      20    0 2548 740 676 S  0.0 0.1  0:00.00 /usr/sbin/acpid
531 avahi      20    0 8532 3088 2760 S  0.0 0.3  0:00.07 avahi-daemon: running [vm-ubuntu20.local]
532 root      20    0 19352 3032 2824 S  0.0 0.3  0:00.00 /usr/sbin/cron -f
660 root      20    0 337M 13972 11848 S  0.0 1.4  0:00.27 /usr/sbin/NetworkManager --no-daemon
535 root      20    0 337M 13972 11848 S  0.0 1.4  0:02.61 /usr/sbin/NetworkManager --no-daemon
542 root      20    0 49260 12508 7532 S  0.0 1.3  0:00.19 /usr/bin/python3 /usr/bin/networkd-dispatcher --run-startup-triggers
572 root      20    0 222M 9160 6076 S  0.0 0.9  0:00.00 /usr/lib/policykit-1/polkitd --no-debug
657 root      20    0 229M 9160 6076 S  0.0 0.9  0:00.61 /usr/lib/policykit-1/polkitd --no-debug
544 root      20    0 229M 9160 6076 S  0.0 0.9  0:02.99 /usr/lib/policykit-1/polkitd --no-debug
592 syslog    20    0 219M 3732 2972 S  0.0 0.4  0:00.04 /usr/sbin/rsyslogd -n -iNONE
593 syslog    20    0 219M 3732 2972 S  0.0 0.4  0:00.00 /usr/sbin/rsyslogd -n -iNONE
594 syslog    20    0 219M 3732 2972 S  0.0 0.4  0:00.04 /usr/sbin/rsyslogd -n -iNONE
548 syslog    20    0 219M 3732 2972 S  0.0 0.4  0:00.11 /usr/sbin/rsyslogd -n -iNONE
722 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.09 /usr/lib/snapd/snapd
723 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.07 /usr/lib/snapd/snapd
724 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.00 /usr/lib/snapd/snapd
725 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.00 /usr/lib/snapd/snapd
732 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.00 /usr/lib/snapd/snapd
806 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.24 /usr/lib/snapd/snapd
815 root      20    0 622M 13532 5320 S  0.0 1.4  0:00.10 /usr/lib/snapd/snapd
558 root      20    0 622M 13532 5320 S  0.0 1.4  0:02.27 /usr/lib/snapd/snapd
580 root      20    0 239M 5748 5236 S  0.0 0.6  0:00.00 /usr/libexec/switcheroo-control
  
```

답변 (CPU 사용률 높은 Command 3 개):

1. /usr/bin/gnome-shell
2. /usr/libexec/gvfs-afc-volume-monitor
3. /usr/bin/VBoxClient --clipboard

**[Q 2] Hello, sleep. [배점: 10]**

다음과 같이 동작하는 justSleep.c 프로그램을 작성하세요. 프로그램은 시작 직후 5 초간 sleep 하고, 5 초가 지나면 return 하고 종료합니다. 프로그램을 컴파일 하고, justSleep 이라는 실행 파일을 생성하고, 백그라운드로 실행하세요. & 연산자를 사용하여 백그라운드로 실행하면, 해당 프로그램의 PID 가 화면에 출력됩니다. justSleep 프로세스가 sleep 에서 깨어나기 전에 \$ps 명령을 입력해서, 프로그램의 PID 를 조회하세요. 터미널 화면을 캡처하여 본 문서에 첨부하세요. 소스코드도 첨부문서로 제출하세요.

(터미널 출력 예시) justSleep 의 PID: 17836

```
daniel@danpc1:~/sp2019-2/w07$ ./justSleep &
[1] 17836
daniel@danpc1:~/sp2019-2/w07$ ps
  PID TTY          TIME CMD
 17632 pts/0    00:00:00 bash
 17836 pts/0    00:00:00 justSleep
 17837 pts/0    00:00:00 ps
daniel@danpc1:~/sp2019-2/w07$
```

답변:

```
yeongju@vm-ubuntu20:~/sp2021-2/w07/q02$ ./justSleep &
[1] 5159
yeongju@vm-ubuntu20:~/sp2021-2/w07/q02$ ps
  PID TTY          TIME CMD
 2063 pts/1    00:00:00 bash
 5159 pts/1    00:00:00 justSleep
 5160 pts/1    00:00:00 ps
yeongju@vm-ubuntu20:~/sp2021-2/w07/q02$
```

**[Q 3] randomFork [배점: 10]**

아래와 같이 동작하는 randomFork.c 프로그램을 작성하세요. 프로그램은 명령행 인자로 확률값을 입력 받습니다. 예를 들어, 0.3 를 입력하면 30%의 확률입니다.

프로그램 실행	<code>\$/randomFork 0.3</code>
프로그램 코드	<pre> (pseudo-code) // random fork! While( 30% 의 확률로 ) {     Fork();     PID 와 PPID 출력     IF( 부모 프로세스 이면 )         자식 프로세스를 wait 한 뒤, 리턴 된 PID 출력 } </pre>

한번에 5 개 이상의 프로세스를 생성할 때까지 프로그램을 반복해서 실행하세요. 터미널 화면을 캡처하여 본 문서에 첨부하세요. 소스코드도 첨부문서로 제출하세요.

(터미널 출력 예시)

```

daniel@daniel-VirtualBox:~/SP2018F/W05_process/processID$ ./randomFork 0.5
Prob=0.500000
My PID (6848), My Parent PID (2035)
My PID (6849), My Parent PID (6848)
My PID (6850), My Parent PID (6849)
My PID (6851), My Parent PID (6850)
Waited pid: 6851
Waited pid: 6850
My PID (6852), My Parent PID (6849)
Waited pid: 6852
Waited pid: 6849
My PID (6853), My Parent PID (6848)
My PID (6854), My Parent PID (6853)
Waited pid: 6854
Waited pid: 6853
My PID (6855), My Parent PID (6848)
Waited pid: 6855

```

답변:

```

yeongju@vm-ubuntu20:~/sp2021-2/w07/q03$ ./randomFork 0.3
Prob = 0.300000
My PID (17110), My Parent PID (2063)
My PID (17111), My Parent PID (17110)
My PID (17111), My Parent PID (17110)
My PID (17112), My Parent PID (17111)
Waited pid: 17112
Waited pid: 17111
My PID (17110), My Parent PID (2063)
My PID (17113), My Parent PID (17110)
Waited pid: 17113
yeongju@vm-ubuntu20:~/sp2021-2/w07/q03$

```

#### [Q 4] 프로세스 대시보드 (+ ncurses) [배점: 20]

위의 [Q 2]에서 작성한 프로그램을 확장하여, sleepFor.c 프로그램을 작성하세요. 프로그램은 정수를 인자로 받으며, 주어진 시간만큼 sleep 후 종료합니다. 예를 들어, \$./sleepFor 10 입력 시, 프로그램은 10초간 sleep 후 종료합니다.

아래와 같이 동작하는 processDashboard 프로그램을 작성하세요. Ncurses 라이브러리를 활용하면 좋지만, ncurses 없이 구현해도 됩니다. 프로그램은 하나의 인자를 받을 수 있습니다:

- -r <정수> : 화면에 출력할 실행중인 프로세스 개수 최대값. 예를 들어, -r 3 이라고 입력하면 최대 3 개의 실행중인 프로세스 목록을 화면에 출력합니다. -r 옵션이 주어지지 않으면 기본값인 5 를 사용합니다.

processDashboard 프로그램은 sleepFor 프로세스의 실행만 카운트 합니다. 대시보드 프로그램 화면은 다음과 같이 구성됩니다. -r 5 으로 실행한 예시:

```

<Process Dashboard>

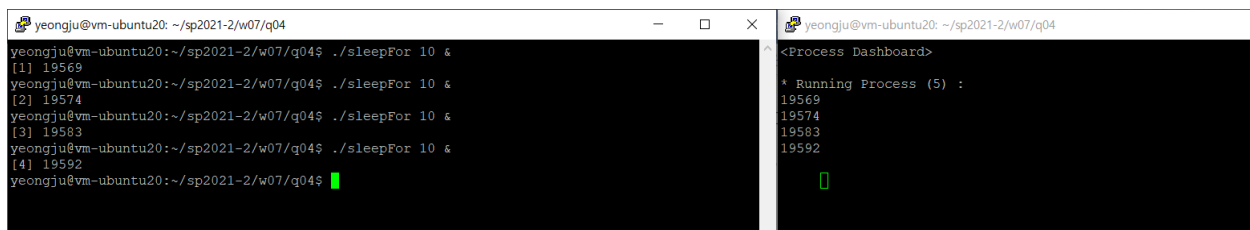
* Running Process (5) :
10000
10004
10005

```

대시보드 프로그램은 종료하지 않고 계속해서 동작하며, sleepFor 프로세스가 실행되면 Running Process 에 해당 프로세스의 PID 를 출력하여 보여주며, 해당 프로세스가 종료하면 해당 PID 값을 제거합니다. 표시할 PID 개수가 지정된 최대값을 초과하면, 실행중인 sleepFor 프로세스 중 우선순위 없이 최대 개수의 PID 를 화면에 출력합니다.

대시보드 프로그램을 실행하고, Running Process 항목에 최소 3 개 프로세스가 나타나도록 sleepFor 프로그램을 구동하세요. 터미널 화면을 캡처하여 본 문서에 첨부하세요. 소스코드도 첨부문서로 제출하세요.

답변:



```

yeongju@vm-ubuntu20: ~/sp2021-2/w07/q04
yeongju@vm-ubuntu20:~/sp2021-2/w07/q04$ ./sleepFor 10 &
[1] 19569
yeongju@vm-ubuntu20:~/sp2021-2/w07/q04$ ./sleepFor 10 &
[2] 19574
yeongju@vm-ubuntu20:~/sp2021-2/w07/q04$ ./sleepFor 10 &
[3] 19583
yeongju@vm-ubuntu20:~/sp2021-2/w07/q04$ ./sleepFor 10 &
[4] 19592
yeongju@vm-ubuntu20:~/sp2021-2/w07/q04$

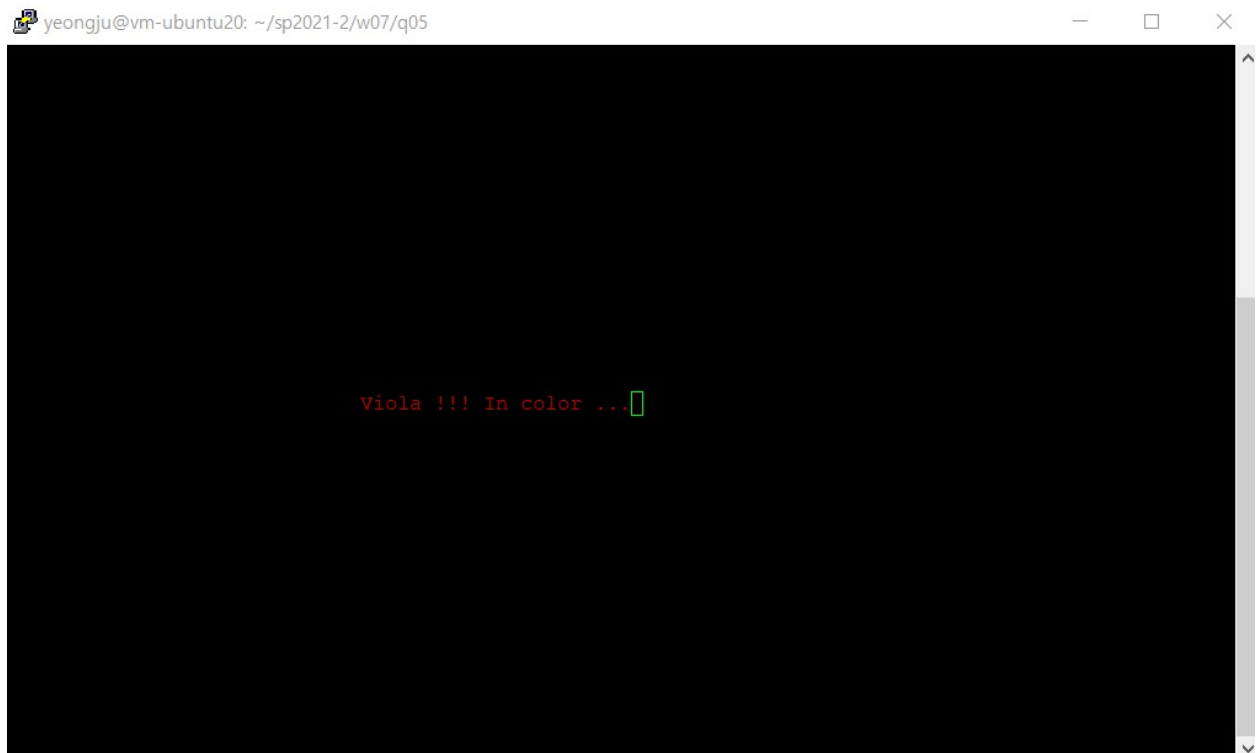
<Process Dashboard>
* Running Process (5) :
19569
19574
19583
19592
  
```

#### [Q 5] Ncurses 컬러 컬러! [배점: 10]

첨부된 colorMain.c 프로그램을 실행하세요. 터미널 화면을 캡처하여 본 문서에 첨부하세요.

(첨부파일 위치: <https://github.com/overegoz/sysprog/tree/master/w07-lab>)

답변:



A terminal window titled 'yeongju@vm-ubuntu20: ~/sp2021-2/w07/q05' with standard window controls. The terminal background is black, and the text 'Viola !!! In color ...' is displayed in red. A green cursor is positioned at the end of the line.

```
yeongju@vm-ubuntu20: ~/sp2021-2/w07/q05  
Viola !!! In color ...
```

**[Q 6] Ncurses 키보드 입력 [배점: 10]**

첨부된 keyboardMain.c 프로그램을 실행하세요. 터미널 화면을 캡처하여 본 문서에 첨부하세요.

(첨부파일 위치: <https://github.com/overegoz/sysprog/tree/master/w07-lab>)

답변:



```

yeongju@vm-ubuntu20: ~/sp2021-2/w07/q06
Use arrow keys to go up and down, Press enter to select a choice
(Charcter pressed = 113 or 'q')

      lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
      x                                x
      x Choice 1                      x
      x Choice 2                      x
      x Choice 3                      x
      x Choice 4                      x
      x Exit                          x
      x                                x
      x                                x
      mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj

```

```

yeongju@vm-ubuntu20:~/sp2021-2/w07/q06$ ./keyboardMain
You chose choice 3 with choice string Choice 3
yeongju@vm-ubuntu20:~/sp2021-2/w07/q06$ ./keyboardMain
You chose choice 5 with choice string Exit

```

### [Q 7] 키보드 + 컬러 [배점: 10]

위의 [Q 6]번 문제에서 사용한 코드에 아래의 내용을 추가하세요.

- 두개의 COLOR\_PAIR 생성
  - `init_pair(1,COLOR_WHITE,COLOR_BLACK)`
  - `init_pair(2,COLOR_BLACK,COLOR_WHITE)`
- 숫자 1 번 버튼 클릭 시, 1 번 color pair 를 사용하여 메뉴 출력. 1 번 color pair 는 default 값임.
- 숫자 2 번 버튼 클릭 시, 2 번 color pair 를 사용하여 메뉴 출력

참고: `wattron/wattroff` 함수는 특정한 window 의 속성을 on/off 합니다.

터미널 화면을 캡처하여 본 문서에 첨부하세요. 총 2 개의 캡처화면을 첨부해야 합니다 : 1 번

버튼 클릭 후 화면 한 개, 그리고 2 번 버튼 클릭 후 화면 한 개. 소스코드도 첨부문서로 제출하세요.

답변:

yeongju@vm-ubuntu20: ~/sp2021-2/w07/q07

Use arrow keys to go up and down, Press enter to select a choice

```

lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
x                                     x
x Choice 1                           x
x Choice 2                           x
x Choice 3                           x
x Choice 4                           x
x Exit                               x
x                                     x
x                                     x
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj

```

Use arrow keys to go up and down, Press enter to select a choice

```

lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
x                                     x
x Choice 1                           x
x Choice 2                           x
x Choice 3                           x
x Choice 4                           x
x Exit                               x
x                                     x
x                                     x
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj

```


**[Q 8] Ncurses 키보드 입력 [배점: 10]**

첨부된 2ncMove.c 프로그램을 실행하세요. 터미널 화면을 캡처하여 본 문서에 첨부하세요.  
아래의 질문에 답하세요.

1. 방향키(위, 아래)의 역할은 무엇인가?
2. 'd' 및 'r' 키의 역할은 무엇인가?
3. 'q' 키의 역할은 무엇인가?
4. 'w' 및 'x' 키의 역할은 무엇인가?

(첨부파일 위치: <https://github.com/overegoz/sysprog/tree/master/w07-lab>)

답변:

 yeongju@vm-ubuntu20: ~/sp2021-2/w07/q08

```
> pizza
oil
apple

Info: Italian pizza from Pusan
Box is null
```

질문 1: 메뉴 선택

질문 2: d(메뉴 삭제), r(메뉴 추가)

질문 3: 종료

질문 4: w(박스 생성), x(박스 미생성)

끝! 수고하셨습니다 ☺