

課題 No.5 の解答例

課題 5-1 プロセス数を調べる.

- a. システム内の全プロセス数を求めなさい.

```
ps オプション
a: 他人のプロセスも表示
u: 詳細な情報を表示
x: 端末に属さないプロセスも表示
$ ps ax | wc
    536    3228    75717
ps コマンドの出力には1行のヘッダが付くので 536-1=535 プロセス
```

- b. システム内の全自分プロセス数を求めなさい.

```
ps が全自分プロセスを表示する
$ ps x | wc
    317    2018    55223
ps コマンドの出力には1行のヘッダが付くので 317-1=316 プロセス
```

```
ps はロングフォーマットでシステムの全プロセスを出力し、
例えば sigemura で始まる行だけを選びカウントする.
$ ps aux | grep ^sigemura | wc
    314    3883    71726
grep プロセス分多いはずなのに逆に 2 少ない.
ps x では root の login プロセスが 3 つ含まれているが今回は含まれない.
```

課題 5-2 二つのターミナルを用いてタイマープログラムプロセスを観察する.

1. タイマープログラムの PID と STAT の確認

```
% ps u
USER      PID   %CPU %MEM    VSZ   RSS  TT  STAT STARTED     TIME COMMAND
sigemura 17419   0.0  0.0 407972224  1296 s001 S+   11:04AM  0:00.01 ./timer
sigemura 17367   0.0  0.0 408300576  3840 s001 S    11:02AM  0:00.09 -zsh
sigemura 17026   0.0  0.0 408169504  3616 s000 S    10:28AM  0:00.07 -zsh
```

timer の PID は 17419, STAT は S+ (短時間のスリープ)

2. タイマープログラムを一時停止させる.

```
% kill -TSTP 17419
```

もう一方のターミナルの時計表示が止まった.

3. タイマープログラムの STAT を再度確認

```
% ps u
USER      PID   %CPU %MEM    VSZ   RSS  TT  STAT STARTED     TIME COMMAND
sigemura 17026   0.3  0.0 408169504  3376 s000 S    10:28AM  0:00.11 -zsh
sigemura 17419   0.0  0.0 407972224   816 s001 T    11:04AM  0:00.10 ./timer
sigemura 17367   0.0  0.0 408300576  3376 s001 S+   11:02AM  0:00.09 -zsh
```

STAT が T (一時停止状態) に変化した.

4. タイマープログラムを再開させる.

```
% kill -CONT 17419
```

もう一方のターミナルの時計表示が再開された.

5. タイマープログラムを終了させる.

```
% kill -INT 17419
```

「課題の準備」でタイマープログラムは`^C`で画面を消して終了していた. `^C`と同じシグナルを送ると同じ終わり方をする.

課題 5-3 以下を一つのターミナルで行う.

1. タイマープログラムを(起動した後,)一時停止する.

```
% ./timer  
... 時計の表示がされる...  
^Z  
zsh: suspended ./timer  
%
```

`^Z`を入力すると一時停止しプロンプトが表示される.

2. 一時停止したタイマープログラムを再開する.

```
% fg  
... 時計の表示が再開される...
```

`fg`と入力するとプログラムがフォアグラウンドで実行再開する.

3. タイマープログラムを終了させる.

```
... 時計の表示がされている...  
^C  
... 画面がクリアされる...  
%
```

`^C`を入力するとプログラムが終了しプロンプトが表示される.