

## プロセス制御

( 以下の定義は大雑把なもの )

- プログラムとは  
CPUが実行できる機械命令の列  
たとえば /bin/ls というファイル
- プロセスとは  
メモリ上に貼り付けられたプログラムをコンピュータが実行している状態にあるもの

40

## プロセスに関する概念

- プロセスの保護  
許可を出さない限り他のプロセスの影響を受けない
- 資源割り当て  
プロセスはメモリ、ディスク、ディスプレイ、キーボードといった資源を割り当てられる
- ユーザの代理  
コンピュータの中で活動する

41

## プロセスの操作

- 新しくプロセスを作る、プログラムを実行する
- 動いているプロセスを調べる
- 動いているプロセスを殺す（英語でもkill）
- 動いているプロセスを一時的に停止させる
- 停止しているプロセスの実行を再開する
- プロセスの優先順位を変える
- プロセスの実行の様子を調べる

42

## シェルの動作

プロンプトを表示し、ユーザから入力されたコマンドを解釈実行しているシェルも1つのプロセスである。シェルは以下のように動作する。

- 1 プロンプトを表示し、入力を待つ
- 2 ユーザからの入力を解釈し、コマンドのプロセスを生成し、制御を渡す  
（この場合、シェルを親プロセス、コマンドのプロセスを子プロセスと呼ぶ）
- 3 子プロセスの終了を待ち、1へ戻る

43

## プロセスの観察

- ps コマンドを使う
- | PID | TTY   | TIME  | CMD  |
|-----|-------|-------|------|
| 123 | pts/3 | 00:00 | bash |
| 134 | pts/3 | 00:00 | ps   |

1行に1プロセスを表示  
左からプロセスID、端末名、CPU時間、起動した時のコマンド

44

## psコマンドのuオプション

資源の割り当て状況を示す

- %CPU --- 過去1分間に利用したCPU時間の割合
- %MEM --- 実際に消費しているメインメモリの大きさの割合
- RSS ( resident set size ) --- 実際に消費しているメインメモリの大きさ
- START --- プロセスが生成された時刻

45

## psコマンドのオプション

- PPID ( parent PID ) --- 親プロセスを示す
- PRI ( priority ) --- 優先順位
- STAT ( status ) --- プロセスの状態
  - R ( runnable ) --- 実行可能状態
  - D ( disk ) --- ディスク入出力の終了待ち
  - S ( sleep ), I ( idle ) --- キーボードや他のプロセスからの入力待ち
  - Z ( zombie ) --- すでに終了していて終了処理の完了待ち

46

## psコマンドの他のオプション

- a --- 他のユーザのプロセスも表示
- x --- 制御端末のないプロセスも表示
- ax --- すべてのプロセスの表示  
( オプションaとxの組み合わせ )

47

## プロセスを強制終了する

- C-c ( まずこれを試そう! )
- killコマンド  
引数はプロセスID ( 例 kill 2312 )  
これで死なない時は -KILL オプションを  
つける ( 例 kill -KILL 2312 )