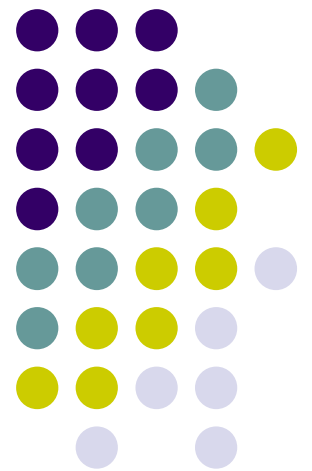


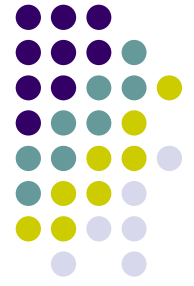
# 第1回 BASHゼミ

---

- ・始めに
- ・スクリプトファイルの作成
- ・補足

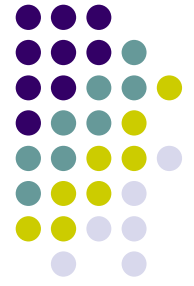
発表者: 鈴木 朋央





# 始めに

- スクリプト言語にはPerl,Ruby,Python,JavaScriptなどがある
- 長所
  - ・コンパイルが不要
  - ・比較的適当に書ける
  - ・動的型付けである
- 短所
  - ・実行速度が遅い
  - ・Cのようなローレベルの操作ができない
  - ・最適化が困難である

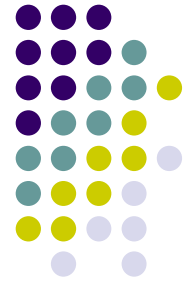


## 始めに(2/2)

- 実行速度は遅いが、記述の容易さにおいて融通が利く！



- プロトタイプや、ちょっとしたプログラムに向く
- BASHスクリプトは基本的にBASHコマンドプロンプトでズラズラ書けるものをファイルに保存したものである。ので、日常生活を送る上で必要である



# スクリプトファイルの作成

- ファイルの先頭に、BASHスクリプトであることを宣言する必要がある

```
#!/bin/sh
```

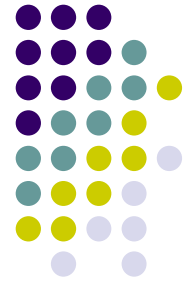
- スクリプトはなるべくexit文で終わるようにする
- 正常終了ならexit 0,異常終了ならexit 1である

なぜこうするかはif文の章で・・・



## 補足 ~ bashとは ~

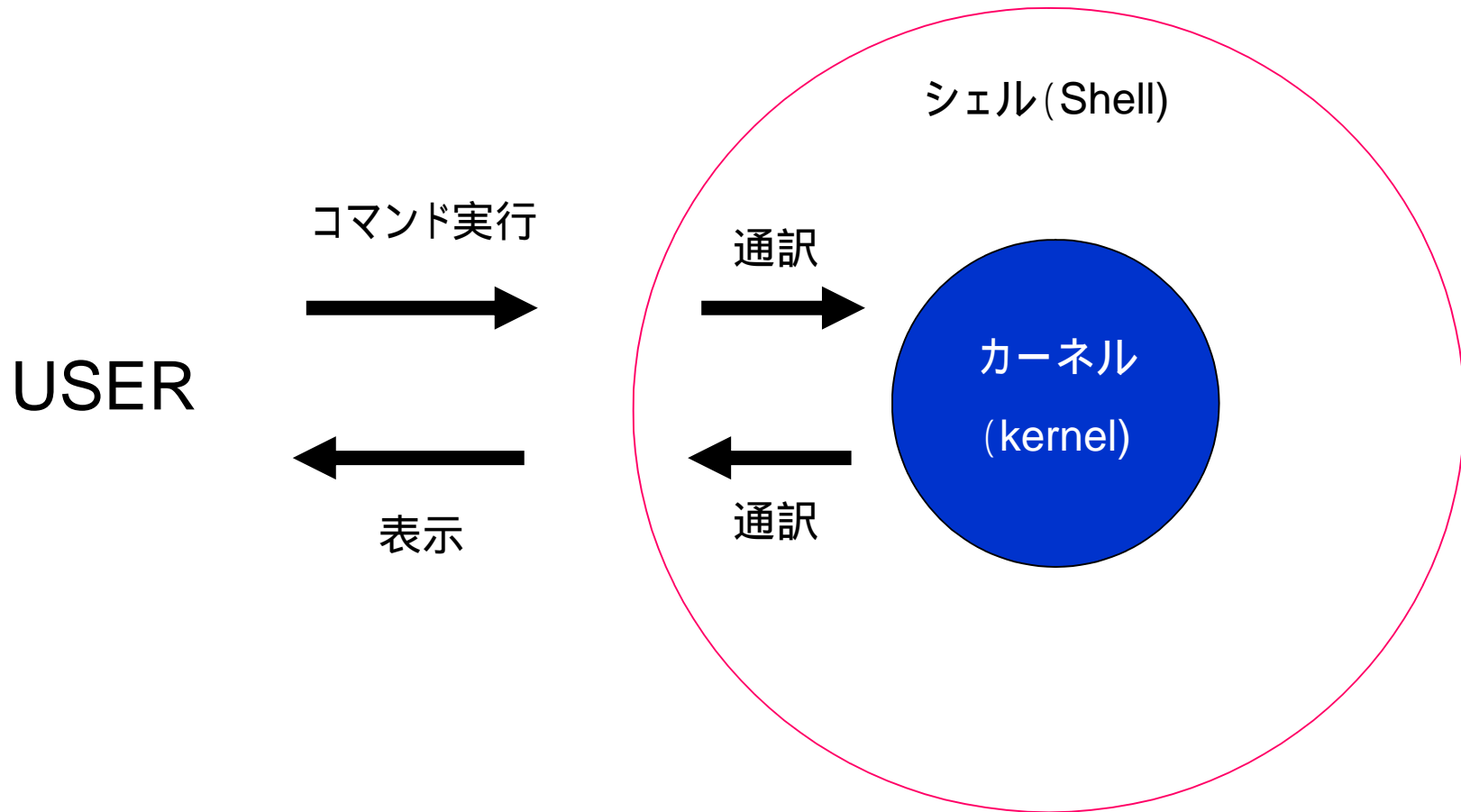
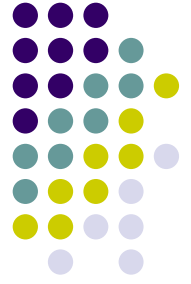
- BASH(Borune Again Shell)とは、数あるシェルの内の1つで Bourne シェルを拡張したもの
  - Bシェル系 : sh,ksh,bash,(zsh)
  - Cシェル系 : csh,tcsh,(zsh)
- Bourne Shell を基本として C Shell や Korn Shell の機能を取り入れ、GNUプロジェクトの一部として開発された。
- Linuxの標準シェル

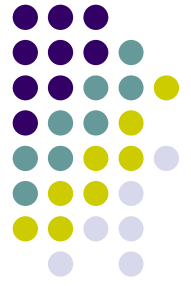


## 補足～シェルとは～

- カーネルと人間の通訳的存在
- 1つの実行プログラムである
- コマンド・インタプリッタとしての役割  
プロンプトを出し、ユーザの入力したコマンド・ラインを解釈して、プログラムを実行し、正常もしくは不正常に終了したら再びプロンプトを出す
- プログラム言語としての性格  
シェルの約束に従って、スクリプトを書き、それをシェルに解釈させてプログラミングを行うことが可能である

# 補足 ~ シェルとカーネル ~





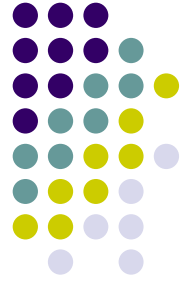
## 補足 ~ Linuxのカーネル ~

- Linux: モノリシックカーネル

入出力機能やネットワーク機能、デバイスのサポートなどOSの一般的な機能をカーネル内部に実装する手法

- Mac OS, TRONなど…: マイクロカーネル

対して、上記の機能をカーネル内に含まず外部モジュール化する等で実装する手法



# 練習問題 1

- 各種シェル (sh, ksh, bash, zsh, csh, tcsh などから適当に) について、各々の特徴を挙げなさい。またその際、bash以外のシェルを用いるとしたらどれを選択するか、その有用性も挙げて記述しなさい。