

計算機ソフトウェア'94 # 2

久野 靖

1994.5.17

3 Tcl 言語

3.1 スクリプト、コマンド、語

- Tcl 言語のプログラムはコマンドの列。
- 1行1コマンドまたは;で区切ってもよい。

3.2 コマンドの評価

- Tcl インタプリタはまず語に分解し、それから評価する。
- {と} はリストを作る。
- 「余計な評価」は頼まない限り起こらない。(これが難しい)

3.3 変数置換

- 語に分解している時\$○があると変数の値で置き換え。

3.4 コマンド置換

- 語に分解している時 [] があると中をコマンドとして実行し、その結果の値で置き換え。

3.5 バックスラッシュ置換

- 特殊文字を引用する。また継続行にも使う。

3.6 ダブルクォート

- ダブルクォートの中では変数置換などが起きる。

3.7 {と}による引用

- {と}の中ではすべて「そのまま」取りこまれる。
- 入れ子になってもよい。

3.8 コメント

- #から行末まではコメント。

3.9 正常終了と例外

3.10 置き換えについての補足

- スキャンして語に分解するとともに置き換え…1回だけ。
- 再度スキャンして欲しければ eval を使う。

4 変数

4.1 単純変数と set

- 変数名はかなり自由につけられる。
- set した時になれば変数が作られる。
- 値はいつでも文字列として格納。

4.2 配列

- 名前 (添字式) で配列になる。(連想配列)

4.3 変数置換

- 変数 (変数) のようにして参照できる。

4.4 unset

- unset で変数をなくせる。

4.5 多次元配列

- 多次元配列はないが、`$a($i,$j)` でよい。

4.6 incr と append

- 「incr 変数 値」で増やせる。
- 「append 変数 値」で値を追加できる。

4.7 その他…

5 式

5.1 数値オペランド

- 整数、実数、16進、8進

5.2 演算子と順位

- 表 5.2 参照。

5.3 数学関数

- `expr`、`sin`、`cos` などはある。

5.4 置換

- これまでの変数置換に加え、`expr` コマンドは自分でも置換をやる。
- `while` などの条件部もそのつど置換される。これは重要。

5.5 文字列操作

- 変数の値はいつも文字列であるのに注意。
- 条件として文字列比較演算子が見える。

5.6 型と変換

- 数値演算が必要になるとそのつど数値に変換。

5.7 精度

- 内部では `double` で計算。
- 文字列に戻す時の精度は `tcl_precision` 変数で指定できる。

6 リスト

6.1 リストの基本と `lindex`

- リストは任意個の要素の並び。要素がまたリストでもいい。
- `lindex` で要素を取り出せる。最初の要素は 0 番。

6.2 リストの生成

- `concat` はリストを連結する。
- リストは与えた引数を 1 つのリストとして返す。
- `llength` はリストの長さ。

6.3 リストの変更

- `linsert` でリストの中に要素を挿入できる。
- `lreplace` で部分的に置き換えられる。
- `lrange` で部分リストを取り出せる。
- `lappend` で複数要素を追加できる。

6.4 リストの探索

- `lsearch` でリストの中に指定した要素があるか、また何番目にあるかを調べられる。

6.5 リストの整列

- `lsort` でリストを並べかえられる。

6.6 文字列とリストの行き来

- `split` を使うと文字列を区切り文字で分けてリストにできる。
- `join` を使うとリストを区切り文字でつながった文字列にできる。

6.7 リストとコマンド

- コマンドとして使うものはリストにしておくのがよい。
- 文字列だとそれを最初に分解する時置き換えとスキャンがなされるので、うまくいかないことがある。
- 動的にコマンドを組み立てるには `list` を使えばよいことになる。

7 制御フロー

7.1 `if`

- 普通の `if`。例を参照のこと。 `else if` でなく `elseif`。

7.2 ループ: `while`, `for`, `foreach`

- 例を参照。 `for` は C の `for`。 `foreach` はシェルの `for`。

7.3 ループ制御: `break`, `continue`

- C と同じ。

7.4 `switch`

- 書き方が2つある。

7.5 eval

- 既に出てきた通り。

7.6 source

- 他のファイルから読み込み。

8 手続き

(略)

9 エラーと例外

(略)

10 文字列操作

10.1 ファイル名マッチふう機能

- 「string match パターン 文字列」。

10.2 正規表現

- 「regexp パターン 文字列」。

10.3 正規表現による置換

- 「regsub パターン 文字列 置き換え 変数」。

10.4 format

- printf や sprintf のようなもの。

10.5 scan

- sscanf のようなもの。

10.6 文字の取り出し

- string index : 文字の取り出し。
- string range : 部分文字列の取り出し。

10.7 探索と比較

- string first/last : 最初/最後から探す。
- string compare : 文字列比較

10.8 長さ、大文字小文字、切捨て

- string length/toupper/tolower/trim

11 ファイルアクセス

11.1 ファイル名

- Unix の普通の構文。

11.2 基本 I/O

- 例題参照。

11.3 出力バッファリング

- 必要なら flush を。

11.4 ランダムアクセス

- seek/tell/eof。

11.5 カレントディレクトリ

- pwd/cd。

11.6 ファイル名展開

- glob でシェルの展開と同様にできる。

11.7 ファイルの情報

- ls -l などと同様の情報が取れる。file コマンド。

11.8 システムコールでのエラー