

# プログラミング基礎

## 第7回 フォルダ、文字列処理

# 今日の内容

- 文字列処理(先週のつづき)
- 複数ファイル・フォルダ

# 文字列処理： つなげる

文字列 1 は「こんにちは」

文字列 2 は「さようなら」

「{文字列 1} {文字列 2}」を表示  
文字列 1 & 文字列 2 を表示

文字列 1 に文字列 2 を追加  
文字列 1 を表示

# 文字列処理：整形する

数は123456

数を通貨形式で表示

桁数は10

数を桁数でゼロ埋めして表示

数を桁数で文字列センタリングして表示

数を桁数で文字列右寄せして表示

# 文字列処理： 一部分を抜き出す

文は「たけやぶやけた」

「{文}の文字数は{文の文字数}」を表示

文の0から3文字抜き出して表示

文の4文字左部分を表示

文の5文字右部分を表示

# 文字列処理： 1文字ずつ調べる

文は「たけやぶやけた」

数は0

文の**文字列分解**を反復

数 = 数 + 1

「{数}文字目：{対象}」を表示

- 文字列分解：1文字ずつ入った配列になる
  - 配列の命令が使えるようになる

# 文字列処理： 文字列の中から探す

文は「たけやぶやけた」

文の「や」を文字検索を表示

文の「や」を4から文字検索を表示

文の「やぶ」を文字検索を表示

文の「ぶけ」を文字検索を表示

- 文字検索：何文字目に出てくるかを調べる
- 出てこなかったときは0

# 課題

- 回文かどうか判定するプログラムを作ろう
  - 「回文」は前から読んでも後ろから読んでも同じ文になる文のことです
    - 例: たけやぶやけた (竹やぶ焼けた)
  - 文章はユーザに尋ねるようにします (ひらがなで入力します)

# ヒント

- 方針A: 反転した文字列を作る
  1. 「文字列分解」「配列逆順」「配列結合」で反転
  2. 元の文字列と等しい？
- 方針B: 左右の1文字を調べる
  1. 一番左の文字と一番右の文字が等しい？
  2. 等しかったら両端を除いた文字列を作る  
(等しくなかったら回文ではない)
  3. 1～2を繰り返す  
(空になったら終わり)

# フォルダの中身を反復

マイドキュメントの**ファイル**列挙を反復  
対象を表示

「{マイドキュメント}\*.**nako**」の**全ファイル**列挙を反復  
対象を表示

マイドキュメントの**フォルダ**列挙を反復  
対象を表示

- 全ファイル列挙・全フォルダ列挙を使うと、子フォルダの中まで反復できる
- 拡張子で限定 「{フォルダ}\*.**拡張子**」

# 例：フォルダ内のファイル一覧

フォルダはマイドキュメントで**フォルダ選択**

「{フォルダの**長いファイル名取得**}にあるファイル一覧」を表示

「{フォルダ}\*.\*.nako」の全ファイル列挙を反復  
対象の長いファイル名取得の**ファイル名抽出**を表示

- 「フォルダ選択」でダイアログが出せる

# 練習

- ファイル一覧表示で、ファイルの行数と文字数を表示するようにしてみよう
- ヒント
  - 「毎行読む」を使って行数と文字数を数える
- 第7回の課題はありません
  - 第6回課題「回文かどうか判定する」だけ提出してください

フォルダはマイドキュメントでフォルダ選択

「{フォルダの長いファイル名取得}のファイル一覧」を表示

「{フォルダ}\*.\*.nako」の全ファイル列挙を反復  
対象のファイル情報表示

●ファイル情報表示（ファイルの）

ファイルの長いファイル名取得のファイル名抽出を表示

行にファイルを毎行読む

行合計は0

文字合計は0

行を反復

行合計に1を直接足す

文字合計に対象の文字数を直接足す

「行数：{行合計}、文字合計：{文字合計}」と表示