

ソフトウェア概論 A/B

数学科 吉開範章, 渡辺俊一 (栗野 俊一)

2010/10/22 ソフトウェア概論

お知らせ

□ソフトウェア概論 **B** は原則、栗野が担当します

□資料

<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino/2010/soft/soft.html>

本日の概要: 文字列の基本

□ 講義内容

○ 文字列の基本 (9 章 : p.208 p.223)

▷ 9-1 文字列とは (p.208 p.213)

▷ 9-2 文字列の配列 (p.214 215)

▷ 9-3 文字列の操作 (p.214 223)

□ 課題

○ 演習 9-4 (p.217)

○ 演習 9-8 (p.219)

○ 演習 9-10 (p.223)

前回までの復習

□ 再帰的関数

- 関数 f の本体に f が現れる事(直接的再帰)
 - ▶ f が呼び出す可能性のある関数の中に f が入っている場合もある(間接的再帰)
- 数学の世界では、「(数学的)帰納法」 / 「帰納的関数」と呼ばれる
 - ▶ 数学では「普通」の考え方

□ 再帰呼び出し

- 関数 f が最初に呼出された後、その結果として再び f が呼出される事

□ 再帰的関数の定義方法

- 通常の関数と基本的には同じ
 - ▶ 関数の本体の中で他の関数を呼出してもよい(例え、それが自分自身であっても..)
 - ▶ 「記法」としては、特に区別はない(構文的な区別ではなく意味的な区別)

□ 再帰的関数の定義パターン

- 再帰的関数は一旦呼出されると、繰り返し呼出される
 - ▶ 呼出される再帰的関数の引数は、「小さくなって」行く
- 数学的帰納法と同様、「二つの場合」の記述が必要
 - ▶ 「最小」の場合の処理：ここでは再帰呼び出しを行わない
 - ▶ その他の場合の処理：再帰呼び出しが行われるが、引数を「小さくなっている」必要がある

今週の内容

□ 文字列の基本 (9 章 : p.208 p.223)

- 9-1 文字列とは (p.208 p.213)
- 9-2 文字列の配列 (p.214 215)
- 9-3 文字列の操作 (p.214 223)

文字列

□ 「文字列」とは

○ 文字列とは (0 個以上の..) 「文字の並び」の事

- ▶ 例 1. 「abc」: 「a」, 「b」, 「c」の 3 文字からなる文字列
- ▶ 例 2. 「This is a pen.」: 14 文字からなる (「」の空白も文字)
- ▶ 例 3. 「漢字」: 2 文字からなる (ただし、C 言語では漢字コードに注意)
- ▶ 例 4. 「」: 零文字からなる (空文字)

○ 文字列の属性

- ▶ 文字列の長さ: 何文字から作られているか? (可変長のデータ型)
- ▶ 文字列を構成する文字情報: 文字そのものとその順序

□ C 言語での「文字列」の表現

○ C 言語では「文字列」そのものはない

- ▶ 「文字列型」が存在しない
- ▶ 文字配列(文字型の配列型)があるだけ(配列は固定長)

○ 文字配列を利用した「文字列の表現方法」がある

□ 「C 言語の文字列」の定義: 「文字列」の表現方法

○ 次の様な特別な形をした「文字配列」は、長さ L の文字列

- ▶ 配列の 0 L - 1 番目には、文字列を構成する文字コードが入る
- ▶ 配列の L 番目には、文字列の終りを表す文字コード(EOS/ 「\0」: NUL 文字)が入る

文字列リテラル

□ リテラルとは

○ 何か型の定数を表す「言語上の表現」の事

- ▶ 例 1. 「123」: C 言語で、整数型の 123 を表すリテラル
- ▶ 例 2. 「123.0」: C 言語で、浮動小数点型の 123 を表すリテラル
- ▶ 例 3. 「0.01」、「1e-2」: 共に 0.01 を表す
- ▶ 例 4. 「1」: 整数の「1」
- ▶ 例 5. 「'1」: 「1」 1 文字(の文字コード)
- ▶ 例 6. 「"1"」: 「1」 1 文字からなる長さ 1 の文字列

□ 文字列リテラルとは

○ 文字列を表す為に用いられるリテラル

- ▶ 実際は「文字列」ではなく「文字配列」のリテラル
- ▶ 文字配列は、少なくとも文字列の長さ + 1 のサイズを持つ(NUL 文字が最後に入る)

○ 文字列リテラルの構文

- ▶ 「"」で始まり、文字列のを構成する「文字コード」の列の後に、再び「"」で終る
- ▶ 例 1. 「"ABC"」: 「A」, 「B」, 「C」の 3 文字からなる文字列(文字配列としては、size 4)
- ▶ 例 2. 「"A\nC"」: 「A」, 「改行(\n)」, 「C」の 3 文字からなる文字列
- ▶ 例 3. 「"A"C"」: 「A」, 「"」, 「C」の 3 文字からなる文字列
- ▶ 例 4. 「"A\\C"」: 「A」, 「\」, 「C」の 3 文字からなる文字列
- ▶ 例 5. 「"A\0C"」: 「A」 1 文字からなる文字列(文字配列としては size 4)

文字配列と文字列の関係

□ 「文字列」は「文字配列の特別な形」

○ C 言語上(構文上)では、「文字配列」しかない

- ▶ プログラマが「特別な形の文字配列」を「文字列」として扱う(意味上)
- ▶ 上記の「意味」に C 言語での(不十分ながら..)サポートがある
- ▶ 例: 文字列リテラル / ライブラリ関数

□ 「文字列」へのサポート

○ 文字リテラル(上述)

○ 文字配列の初期化

○ 文字列の入出力

- ▶ 文字列の書式(%s)
- ▶ 無書式の入力危険(gets 使用禁止)

○ 文字操作ライブラリ

- ▶ strlen/strcat/etc.. (strings.h)

文字列処理

□ 文字列処理の基本

- 「文字列」は「特別な形をした文字配列(契約)」という原点に戻る

- ▶ 入力: 文字列の最後には **EOS** が入っていると期待して良い(権利)
- ▶ 出力: 文字列の最後には、**EOS** が入っていると期待されている(義務)

□ 文字列処理の考え方

- 文字配列の処理

- ▶ ただし、長さが可変なので **EOS** を頼りに処理

□ 文字列の再帰的定義

- 「」 空文字列は文字列

- **A** が文字で **s** が文字列なら **As** も文字列

課題

□ 課題

- 9-1 文字列とは (p.208 p.213)
- 9-2 文字列の配列 (p.214 215)
- 9-3 文字列の操作 (p.214 223)

おわり

終了