

ソフトウェア概論 B

数学科 吉開範章, 渡辺俊一 (栗野 俊一)

2009/12/11 ソフトウェア概論

おしらせ

□資料

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

□ネットワークケーブルは...

○本日は雨のため、準備できませんでした...

□ソフトウェア概論Bの定期試験

○1月29日の1限目

□雨と事故で、電車が遅れているようです

○すこし遅く始めます

○本日の資料は上っていますので、参照してみてください

前回のまとめ：文字列とポインター

□ C 言語の文字列

- 言語上は、「文字列型」は存在しない
 - ▶ 文字配列 (`char string[] + EOS`) で実現
 - ▶ 「番兵」を利用
- 「文字列」リテラルは (`char *`) 型のポインター定数値

□ 基本的な文字列操作

- ライブラリ関数を利用 (Text p.260 265)
 - ▶ 基本的な関数は覚えておく必要がある
 - ▶ `strlen` : 文字列の長さを求める
 - ▶ `strcpy` : 文字列のコピーを行う (領域が重なる場合は未定義)
 - ▶ `memmove` : 文字配列の内容の一部をコピー / 領域が重なる場合に利用
 - ▶ `strstr` : 文字列の中から部分文字列を探す

本日の概要：構造体の導入

□ 配列内のデータのソーティング (Text p.268-271)

- 配列の中のデータを大きさの順に並び換えたい

- ▶ バブルソート：比較的簡単なアルゴリズムの例

□ 構造体の導入 (Text p.272-279)

- 構造体の必要性
- 構造体とは
- 構造体の書き方

□ 課題

- ソーティングの確認：float 型配列のソート
- 構造体の利用：キーボードから構造体へのデータ入力

□ 次回(予定)

- 構造体の応用 (Text p.280-287)
 - ▶ 構造体配列のソーティング
 - ▶ 複雑なデータ構造の実現

ソーティング (触りだけ..)

□ ソーティング (Sorting:整列) とは？

- 順序付け可能なデータの集まりを順番良く並べる(整列する)

 - ▶ 順序付可能なデータ：身長 (成績/名前/etc..)

 - ▶ データの集まり：配列

 - ▶ 順序良く：配列の添字の順とデータの大きさの順が揃っている

- なぜ、「並べ替え」が「分類(Sort)すること」になる？

 - ▶ 大きさの順に並べれば、同じ大きさの物が連続して並ぶ：分類

- 計算機で、大量のデータを扱う際の「基本的な操作」の一つ

□ ソーティングアルゴリズム

- ソーティングを行う事が判っている「手順」の事

 - ▶ 様々な物がある [cf. 参考文献 [Link](#)]

 - ▶ 詳しくは来年度の「アルゴリズム概論」を履修の事

- 今回はバブルソートを紹介

バブルソート（泡立ち法）

□バブルソート (bubble sorting) とは

- ソーティングアルゴリズムの一つ

- 特徴

- ▶隣り合ったデータを比較する (単純交換法)
- ▶配列内の並び替え:与えられた配列を直接変更する
- ▶直観的で解りやすいが遅い (しかし、工夫次第では..)

□バブルソートの手順

- 配列内の隣接したデータを順に調べる [000]

- ▶もし、その二つが整列状態でなければ、そのデータを交換する

- 改良 [001]

- ▶途中で、交換が一回も起きなければ、そこで止めてよい
- ▶最後に交換が起きた場所(k)までやればよい

構造体

□ 構造体とは

- 関連するデータをまとめたもの [002]

- ▶ 複数のデータからなる複合データを表現する

□ 構造体の宣言 [003]

- **struct** タグ名 { メンバ宣言 };

- ▶ 構成するメンバの名前と型を指定する

□ 構造体型の変数の定義

- **struct** タグ名 変数名;

- ▶ 複数の変数をまとめて定義するイメージ

□ メンバへ参照 (「. (ドット)」 演算子)

- 「変数名.メンバ名」で、そのデータの構成要素を参照

```
struct point { int x; int y; }
```

```
sturct point p1;
```

↓

```
int p1.x; /* イメージ：実際にはできない */
```

```
int p1.y;
```

構造体の利用

- 構造体型変数の初期化
 - メンバ毎に行う [003]
 - 宣言時に一度に行う [004]
 - 代入で一括して行う [005]
- 構造体型変数の値を関数引数に与える
 - 基本は、代入が行われている [006]
 - 関数内で、値を変更するには？
 - ▷ やっぱり、ポインターが必要 [007-009]
 - ▷ (*p).m で参照
 - ▷ p -> m を使うのが普通 [010]
- 構造体型の宣言
 - typedef で「型」を作る [010]

課題

□課題は、次の Web Page の内容を参照してください。

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

おわり

終了