

ソフトウェア概論 A/B

-- ファイル I/O, データ構造 --

数学科 栗野 俊一 / 渡辺 俊一

2013/09/27 ソフトウェア概

伝言

私語は慎むように !!

□ 色々なお知らせについて

- 栗野の Web Page に注意する事

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

□ 廊下側の一列は遅刻者専用です(早く来た人は座らない)

□ 講義開始前に済ませておく事

- PC の電源を入れる
- ネットワークに接続しておく事
- 今日の資料に目を通しておく事

□ 講義前の注意

- 講義前は、栗野は準備で忙しいので TA を捕まえてください

□ やる気のある方へ

- 今日の資料は、すでに上っています
 - ▷ どんどん、先に進んでかまいません

前回 (2013/09/20) の復習

□ 前回 (2013/09/20) の復習

○ 前期(ソフトウェア概論 A)の復習

- ▶ 目標 : プログラムを C 言語で書けるようにする(後期の目標も同じ)
- ▶ 内容 : 操作方法 / 「プログラム」の作成法 / C 言語理解
- ▶ 操作方法 : 基本はできるようになっている
- ▶ 作成方法 : 少ししかやっていないので、後期はこの話が増える
- ▶ C 言語 : 一通りやったがもう一度やるし、取り零しの部分の対応も増える

○ 「C 言語」の内容

- ▶ Hello, World : 取り敢えず動くプログラムが作成、実行できる
- ▶ 関数の利用 : 命令の一部に名前を付ける / それを呼び出せる
- ▶ 三つの制御構造 : 順接(命令を並べる) / 条件分岐(if) / 繰返し(再帰/while)
- ▶ 変数と代入 : 変数宣言による変数の利用と、代入による変数の値の変更
- ▶ データと型 : データには型があり、それによって演算も異なる (int/char/double)

○ 標準 I/O : どこでも利用(標準)できる入出力(I/O)ライブラリ

- ▶ printf : 書式付き、便利出力関数 → 是非、使い熟せるようになりたい
- ▶ scanf : 書式付き、便利入力関数 → 超便利ではあるが、初心者には辛い

お知らせ

□ 本日の予定

○ 標準 I/O ライブラリの続き

▷ fprintf/fscanf, fopen/fclose

○ データ構造

▷ 配列と構造体

□ 本日の目標

○ 演習

▷ 課題の提出

本日の課題 (2013/09/27)

□ 本日 (2013/09/27) の課題

○ 課題 1:

- ▶ ファイル名 : 20130927-1-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : `sscanf` による文字列から情報の取出し
- ▶ ファイル形式 : テキストファイル(C 言語プログラムファイル)

○ 課題 2:

- ▶ ファイル名 : 20130927-2-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : `fopen/fclose/fprintf` によるファイルの書出し
- ▶ ファイル形式 : テキストファイル(C 言語プログラムファイル)

○ 課題 3:

- ▶ ファイル名 : 20130927-3-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 3次元ベクトルの差
- ▶ ファイル形式 : テキストファイル(C 言語プログラムファイル)

標準 I/O ライブラリの利用(再)

□ 様々な標準 I/O ライブラリ

○ 標準入出力(default[指定しないと..] キーボードと画面)

- ▶ printf -- 標準出力(画面)にデータを書き出す
- ▶ scanf -- 標準入力(キーボード)からデータを読み込む

○ scanf(printf) の便利さと危うさ

- ▶ 書式とデータの対応関係は、プログラマが正しく管理する必要がある
- ▶ scanf では「入力の形式」が「予測できない」ので、「危険」
- ▶ *特に* 変数の前に「&」を忘れると、実はとんでもない事に..(後述)

○ sscanf

- ▶ 文字列に対する scanf : 文字列の中から数値を取り出せる
- ▶ 「定形式なもの」に対しては、scanf は大変強力

ファイル I/O

□ ファイルを対象とする I/O

- ファイルは **Open** して利用し、おわったら **Close** する必要がある

 - ▶ `fopen` : ファイルを open する関数

 - ▶ `fclose` : ファイルを close する関数

□ ファイルポインター

- ファイルポインターって？

 - ▶ ファイル情報管理構造体へのポインター(詳しくは後日)

- ファイルを **open** すると、「ファイルポインター」が手に入る

- ファイル(外にある)を内部で扱うには「ファイルポインター」を経由する

□ ファイルへの I/O

- ファイルポインタを使って、`fprintf/fscanf` で行う

ファイルへ入出力(FILE,fopen,fclose)

□ I/O 情報の固定化

- 出力の固定化(保存) → ファイルへの出力
- 入力 of 固定化(再現) → ファイルからの入力

□ リダイレクション(標準入出力の切替)

- コマンドラインで < や > を利用して入出力を切り換える事ができる

□ プログラム内でファイルを指定して入出力

○ fopen で開く(open する)

- ▶ ファイルポインタ値が入手できる
- ▶ 何らかの原因で open (オープン) に失敗するすると NULL 値が返る
- ▶ open をした場合は、成功か失敗かを必ず確認する事 !!

○ fprintf で書込む / fscanff で読み込む

○ fclose で閉じる(close する)

- ▶ !! open したファイルは必ず close する

データの構造

□ 平面上の点を扱う事を考える

- x 座標と y 座標の組で「点」を表現

 - ▶ 点 p_1 の x, y 座標をそれぞれ p_{1x} , p_{1y} で表現してみる

- 点の表示や、距離などは、普通に扱える [010]

- 点そのものを操作する事を考えると..

 - ▶ x 軸, y 軸, 原点に対象な点 .. [011]

 - ▶ 特に関数にすると辛い [012,013]

□ 「点」を表すもの(データ構造)を考える

- 構造体：複数のデータをまとめて扱うようにする仕組み