

# ソフトウェア概論 A/B

-- 総まとめ/模擬試験 --

数学科 栗野 俊一

2012/07/06 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)

### □ 講義開始前にすませておくこと

- PC の電源を入れる / ネットワークに接続しておくこと

▶ 今日の資料に目を通しておくこと

### □ 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

# 前回の復習

---

## □ double 型

- 小数点付きの数 (123.456) を表現する

- ▶ 様々な数学的な関数ができる (sin/cos/tan/etc..)

- ▶ 四則の計算ができる

## □ 代入

- 変数の値を「変更」する事

- ▶ 言い替えると.. ? 「変数は代入が行われると変数の値が変化する」

- 変数に値を代入するには ?

- ▶ 代入文 ( 「変数名 = 式」 ) を使う

- ▶ 式の値が計算された上で、変数に代入される

# お知らせ

---

- 本日の予定
  - 前期のまとめ
  - 模擬試験
- 本日の目標
  - 模擬試験の結果を提出する

# 前回の課題 (2012/06/29)

---

## □ 前回 (2012/06/29) の課題

### ○ 課題 1:

- ▶ ファイル名 : 20120706-1-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 実数値を二つキーボードから読み込み、その四則を計算する
- ▶ 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 : 20120706-2-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 方程式「 $\cos(x)=x$ 」の答を求めよ
- ▶ 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: (前回の課題なので、日付が 0622 である事に注意)

- ▶ ファイル名 : 20120622-3-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 :  $\cos$  のグラフ
- ▶ 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 4: (前回の課題なので、日付が 0622 である事に注意)

- ▶ ファイル名 : 20120622-4-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 素数の判定プログラム
- ▶ 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)

# 本日の課題 (2012/07/06)

---

## □ 本日 (2012/07/06) の課題

- 9999.zip をダウンロードして展開してください
- 問題は question/q??.txt にあります
- 答は answer/a.txt と C 言語のプログラムになります。
  - ▷ qqqq-a.txt : q.11 ~ q.14 の解答
  - ▷ qqqq-00.c ~ qqqq-10.c : q.00 ~ q.10 の解答
- 模擬試験の結果を提出する

# ファイルの入手とインストール

---

## □ ファイルのダウンロード

- 次の本日 (2012/07/06) のページからファイルをダウンロードする

<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino/2012/soft/20120706/20120706.html>

- ダウンロードするファイル

▶ 9999.zip : 展開し「c:\usr\c」に保存する

# 次回の予定

---

## □ 次回(2012/07/13)は試験を行います

### ○ 試験時間は、170 分

▶ 09:10 ~ 12:00 ( 09:30 に出席を取ります )

### ○ 時間内に結果を CST Portal に提出してください

▶ ギリギリに提出しようとするサーバーが対応できない可能性あり

## □ 試験の形式

### ○ ファイルに入った問題をダウンロード

▶ ファイル内に個々に問題が入っているので、それを見て解く

▶ 問題は、各自異なる(ので答も異なる)

▶ 回答は、ファイルの形で、ポータルに upload する

### ○ 持ち込み

▶ Note-PC を含め、なんでも可 (もう一台の PC / 本 / ノート..)

### ○ 禁止事項

▶ 音を出しては駄目 ( 会話不可 / 携帯電話不可 / チャット可 ) / 物の貸し借りは駄目

## □ 質問

### ○ 問題が「変」と思ったら、手を挙げてください

# 総まとめ(1)

---

## □ C Compiler

- C のプログラムを実行ファイルに変換する
  - ▶ エディタで `foobar.c` を作成する
  - ▶ `cc -c foobar.c` で、コンパイル
  - ▶ `cc -o foobar.exe foobar.o` でリンク
  - ▶ `foobar` で実行

## □ Hello, World

- C で作成された「意味のある」「最小」のプログラム
  - ▶ 自分がプログラムを作る時の雛形になる
  - ▶ 沢山の「お呪い」があるが、きにしない
- すでに動いているプログラムを変更して作る

# 総まとめ(2)

---

## □ 関数(引数なし)

- プログラムの一部に名前を付ける
- 関数定義:関数を作る
  - ▶ 「void 関数名(void) { 命令列 }」
- 関数呼び出し: 関数名を利用して、命令を呼び出す
  - ▶ 「関数名();」

## □ 関数引数

- 関数内で、利用する情報(データ)の一部を変数にしてあとから決定する
  - ▶ 仮引数:関数の中で利用される変数
  - ▶ 実引数:関数を呼び出す時の変数に入る値

# 総まとめ(3)

---

## □ 引数宣言

- 最初は「char \*」固定
- 実は、引数宣言は、引数の「型」を指定する
  - ▶ 「char \*」は、「文字列」を表すための型だった

## □ 型

- char \* (文字列), char (文字), int(整数), double(浮動小数点数)
  - ▶ 値の意味 (それが何を表しているか)
  - ▶ 値への演算 (+1 をすると何がおきるか)
- 関数の仮引数の型と、実引数の型は一致している必要がある

# 総まとめ(4)

---

## □ 命令の組立方

### ○ 順接：単に命令を並べるだけ

- ▶ 上から下に、順番に、その命令が実行される
- ▶ cf. `printf("a");printf("b");` => 「ab」と出力される
- ▶ 命令の実行順序が指定できる。

### ○ 条件分岐: 変数の値によって実行する命令を選択する

#### ▶ `if ( 条件式 ) { 真の時 } else { 偽の時 }`

- ◇ 条件式が真(0以外)に「真の時の命令」を実行、それ以外(0の時)は、「偽の時の命令」を行う
- ◇ 実行する命令を変数で「選択」できる
- ◇ プログラム作成時に選択できない場合でも、実行時に `if` 文で選択が可能になる

### ○ ☆ 順接と条件分岐だけではプログラムの長さ以上に命令を実行する事はない

- ▶ 原理的に有限の命令しか実行できない

# 総まとめ(5)

---

## □ 再帰呼び出し

○ 関数を定義する時に、関数の中で、自分自身を利用できる

▷ cf. 階乗の計算をする `fact ( int n )`

◇ `int fact ( int n ) {`

`if ( n < 1 ) {`

`return 1;`

`} else {`

`return fact ( n - 1 ) * n;`

`}`

◇ `}`

◇ `fact ( 3 )`

`=> fact(2) * 3`

`=> ( fact(1) * 2 ) * 3`

`=> ( (fact(0) * 1 ) * 2 ) * 3`

`=> ( (1 * 1) * 2 ) * 3`

`=> 6 (だから fact(3) = 6 となる )`

○ ☆ : 再帰呼び出しを利用すると、同じ命令を何回も実行できる

▷ 原理的に、無限の命令が実行できる

◇ 無限にならない工夫が必要

# 総まとめ(6)

---

## □ 入力と出力

### ○ 出力：画面上に文字列表示する

▶ `printf ( "abc" )` => 「abc」が表示される

▶ `s_print_int ( 123 )` => 「123」が表示される

### ○ 入力：キーボードから文字列を入力すると、それがプログラムに取り込まれる

▶ 取り込む時には、内部の「型」に変換される

◇ `s_input_int()`; キーボードから「123」と文字列いれると 123 という数値に変る

### ○ プログラム(設計)の基本構造

▶ 入力 => 処理 => 出力

# 総まとめ(7)

---

## □ 代入

- 変数の値をプログラムの途中で変更する方法

  - ▷ # 引数で指定された値になり、そのまま

- 「変数名 = 式;」