ソフトウェア概論 A/B

-- 値と型(整数型と浮動小数点数型) --

数学科 栗野 俊一 / 渡辺 俊一

2019/10/11 ソフトウェア概論

伝言

私語は慎むように !!

- 出席パスワード : 20191011
- 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
- VNC Server Address : 10.9.154.227
  - Password : vnc-2018
- 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 の電源を入れておく
  - ネットワークに接続しておく
  - 今日の資料に目を通しておく
- 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

前回の内容

- 前回の内容
  - 変数の値の更新：右辺に左辺と同じ変数を含む代入
    - 以前の変数の値を利用して、今後の変数の値を定める
  - while 構文
    - 「繰返し」を実現するもう一つの手段 (cf. 再帰呼出し)
    - 構文 : while (条件) { 繰返し命令 }
    - 制御変数(while の条件部分で参照され、本体で更新される変数)の更新が必須
  - scanf
    - データをキーボードから入力し、変数に代入する関数
    - 使い方に注意が必要

お知らせ

- 本日(2019/10/11)の予定
  - 表現
    - インプットループ
    - 乱数
    - 値と型と型宣言
    - 整数型/浮動小数点数型
- 本日(2019/10/11)の目標
  - 演習
    - 課題の提出

今週 (2019/10/11) の課題

- 課題 20191011-01:
  - ファイル名 : 20191011-01-QQQQ.c (QQQQ は学生番号)
  - 内容 : キーボードから入力された二つの整数型の値の四則と余りの結果を表示する
  - 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 20191011-02:
  - ファイル名 : 20191011-02-QQQQ.c (QQQQ は学生番号)
  - 内容 : キーボードから入力された二つの浮動小数点数型の四則の結果を表示する
  - 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)

先週 (2019/09/27) の課題

- 先週 (2019/09/27) の課題
  - 課題 20190927-01:
    - ファイル名 : 20190927-01-QQQQ.c (QQQQ は学生番号)
    - 内容： switch 構文
  - 課題 20190927-02:
    - ファイル名 : 20190927-02-QQQQ.c (QQQQ は学生番号)
    - 内容： for 構文
  - ※ ファイル形式は、いずれもテキストファイル(C 言語プログラムファイル)

値と型

- Data(データ) と Code(コード)
  - Data : C 言語のプログラムで「操作の対象」となる物
  - Code : C 言語のプログラムで「操作を行う」命令部分
    - 共に「メモリに記録」されている (将来説明する)
- 値(あたい)と型(かた)
  - Data は、「値」と「型」を持つ
    - 型 : Data がどんな集合の要素で、どの様な演算ができるかを決める属性
    - 値 : 型で指定された集合の要素の一つ (値が同じでも型が異れば異る物)
  - Data の値と型の例
    - 整数値の「1」: 型は「整数型」で、値は 1
    - 文字の「1」: 型は「文字型」で、値は '1'
    - 一文字からなる文字列「1」: 型は「文字列型(仮)」で、値は "1"
  - 型が異れば、できる「演算」も異るし、「結果」も異る
    - 例1 : 9 + 1 = 10, '9' + 1 = ':', "9" + 1 = ""

式

- 式：複数の Data から新しい Data を作る表現
  - 例：1 + 1 → 2 , *"abc" → 'a', "abc" + 1 → "bc"
  - 「式」から Data を得る事ができる (つまり、値と型が得られる)
  - 即値(式)：定数を表す表現
    - 1：整数型の 1 という値を持つ Data を表す「即値式」
    - '1'：文字型の '1' という値を持つ Data を表す「即値式」
    - "1"：文字列型の "1" という値(?)を持つ Data を表す「即値式」
  - 演算子との組み合わせ
    - 整数の四則：12 + 5 → 17, 12 - 5 → 7, 12 * 5 → 60, 12 / 5 → 2
    - 「+」,「-」,「/」,「*」は、二つの整数型の Data から一つの整数型の Data を作る
    - 文字列の先頭：*"abc" → 'a'：型が変化している事に注意 !!
  - 関数呼出し：関数は Data を返す
    - 例 1：getchar()：キーボードから入力された文字に対応する Data
    - 例 2：rand()：呼び出す毎に異る整数が返る

整数型(再)

- 整数型
  - C 言語での整数 (cf. /usr/include/limits.h)
    - 表現できる範囲は限られている
    - 32bit の場合は −2147483648(-2^31) から 2147483647(2^31-1)
  - 宣言：int で行う
  - 計算：四則演算と余りの計算が可能 (+, -, *, /, %)
    - / は整数割り算なので、小数点以下は切捨てになる (余りは % で得る)
  - 比較：大小比較、等号、不等号が使える
    - a > b : a が b より大きい (a < b : a が b より小さい)
    - a >= b : a が b 以上 (a <= b : a が b 以下)
    - a == b : a と b が等しい (「=」でないことに注意 !!)
    - a != b : a と b が等しくない
- 整数型の入出力
  - 入力：printf ("%d", 整数値)
  - 出力：scanf ("%d", & 整数型変数)

浮動小数点数型

- □ double 型
  - ○ 小数点付きの数を表現する
    - ▹ C 言語内での表現：小数点付きの数 (cf. 123.456)
    - ▹ 2.225074*10^{-308} < double の絶対値 < 1.797693*10^{308}
  - ○ 宣言：double で行う
  - ○ 計算：様々な数学的な関数が利用できる
    - ▹ sin/cos/exp/log/etc.. (cf. #include <math.h>)
  - ○ (当然) 四則の計算ができる：3.0/2.0 → 1.5 (cf. 3/2 → 1)
  - ○ 比較：大小比較、等号、不等号が使える(整数と同じ)
    - ▹ ただし、浮動小数点数同士の「==」は止めた方がよい
    - ▹ その代りに、比較するニ数の差の絶対値(fabs)を取り、小さな正の浮動小数点数(ε)より小さいかを調べる
- □ 浮動小数点数型の入出力
  - ○ 入力：printf ("%f", 浮動小数点数値)
  - ○ 出力：scanf ("%lf", & 浮動小数点数型変数)

浮動小数点数型と誤差

- C 言語(一般に計算機で)の数値表現は「有限」でしかない
  - 浮動小数点数型の値も、もちろん「有限」な表現
    - 「実数」を正確に表現しているわけではない
- 丸め誤差
  - 実数の本当の値を浮動小数点数で表す必要がある
    - C 言語の中では、「その真値に近い浮動小数点数」に「丸めて」しまう
    - 「丸め誤差」がある
  - 計算誤差
    - 「浮動小数点数」での計算結果も「浮動小数点数」の範囲にはいらない
    - やはり、「丸めて」しまう
- 浮動小数点数型の扱い(誤差を意識して扱う)
  - 浮動小数点数型の値は、常に「誤差」を含む
    - 「誤差」は蓄積される可能性がある
  - 等号比較はしない
    - 「差」が「小さい範囲」にあれば、「等しい」と見做す

CSV ファイル

- CSV ファイルとは
  - 表形式のデータをテキスト形式で表現したもの
    - 「,(カンマ)」でフィールドを、「改行」でレコードを区切る
  - 例

    名前, 年齢, 性別

    鈴木, 35, 男

    田中, 24, 女
  - MS-Excel で、読み書きできる
    - 拡張子 (.csv) のファイルは、Excel で開く事ができる
    - 保存する時に、「CSV 形式」を指定すると、保存できる(データのみ)

CSV ファイルの処理

- CSV ファイルの処理
  - s_csv.zip の内容を利用して、csv ファイルが処理できる
- s_csv.zip の利用方法
  - c:\usr\c\20191011 にダウンロード、全て展開する
    - s_csv フォルダができる
  - ubuntu で、~/c/20191011/s_csv に移動
    - cd ~/c/20191011/s_csv
  - テスト : make でテストする
    - make
- 自分のプログラムの利用
  - make TEST=名前