

ソフトウェア概論 A/B

-- 色々な条件分岐構文 --

数学科 栗野 俊一 / 渡辺 俊一

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く
禁じます

伝言

- 出席パスワード : 20210625
- 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>
- 対面講義の受講者へ
 - 座れる席は QR コードが貼られている席だけです
 - ▷ 携帯電話で、位置情報を送ってください
- 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

お知らせ

ソフトウェア概論 A/B (2021/06/25)

お知らせ

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

お知らせ

□ コロナウィルスのワクチン接種に伴う変更について

○ ワクチン接種会場は、理工学部のキャンパスを利用する事になりました

▶ 接種期間中は、対面講義は無くなり、全ての科目で「オンライン講義」になります

□ ソフトウェア概論の講義

○ ソフトウェア概論は、基本、「オンデマンド講義」になります

▶ 講義の時間に、CST Portal II 経由で動画を公開します

▶ 動画を視聴の上、課題を CST Portal II に提出してください

○ 動画を視聴する時間は自分の好きな時間でかまいません

▶ 生活リズムを保つためできるだけ、時間割通りの時間に受けましょう

前回(2021/06/18)のまとめ

ソフトウェア概論 A/B (2021/06/25)

前回(2021/06/18)のまとめ

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

前回(2021/06/18)の復習

□ 前回(2021/06/18)の内容

○ 講義内容

- ▶ 「入力-処理-出力」
- ▶ s_print.h/s_input.h の使い方
- ▶ 返値(かえりち)のある関数 (1)

○ 演習

- ▶ 課題の提出

今回(2021/06/25)の予定と課題

ソフトウェア概論 A/B (2021/06/25)

今回(2021/06/25)の予定と課題

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

今回(2021/06/25)の予定

□ 出席パスワード : 20210625

○ 出席は CST Portal で取りますが、成績には(残念ながら?)無関係です

▶ 単位を取りたいならば、課題を提出しましょう

□ 本日(2021/06/25)の予定

○ 色々な条件分岐構文

▶ if 構文 again

▶ switch 構文

□ 本日(2021/06/25)の目標

○ 課題の提出

先週 (2021/06/18) の課題

□ 先週 (2021/06/18) の課題

○ 課題 20210618-01

- ▶ ファイル名 : 20210618-01-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 文字の種類に応じて、その種類を表す文字を返す関数
- ▶ 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 20210618-02:

- ▶ ファイル名 : 20210618-02-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 二つの整数値を入力し余りを計算する関数
- ▶ 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)

今週 (2021/06/25) の課題

□ 今週 (2021/06/25) の課題

○ 課題 20210625-01

- ▶ ファイル名 : 20210625-01-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 異なる国の挨拶をする (switch 構文版)
- ▶ 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 20210625-02:

- ▶ ファイル名 : 20210625-02-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 三つの整数値の内、真ん中の数値を返す関数 (if 構文の入れ子)
- ▶ 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)

if 構文 again

ソフトウェア概論 A/B (2021/06/25)

if 構文 again

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

条件分岐 (if 構文)

□ 条件分岐

- 複数の命令の内、条件によって、いずれか一つを実行する仕組み

- ▶ 与えられた状況に併せて、適切な対応を選択する仕組み

- ▶ プログラムの柔軟性を高める仕組み

□ if 構文：条件分岐を実現する C 言語での表現方法

- 構文：if (条件) { 条件の成立時の文 } else { 条件が不成立時の文 }

- ▶ 条件：真偽値(成立/不成立)を求める式

- ▶ 条件の成立時の文 / 条件が不成立時の文：複数の文(命令)が指定できる

- 意味

- ▶ 「条件」が『成立』した時に「条件の成立時の文」を、そうでない時には、「条件が不成立時の文」を実行する

- ▶ どちらか一方だけが実行される (両方実行も両方不実行もなし)

- 「if 『文』」ではなく、「if 『構文』」である理由

- ▶ 「『文』を対象として、『何かをする表現』」なので「メタ表現」

if 構文 アラカルト

□ if 構文の基本 : `if (C) { A } else { B }`

○「基本」は「基本の組み合わせだけで何でもできる」の秀逸

▷ 色々な組み合わせ方を学ぶ必要がある

□ else 節の省略

○ else 節 : if 構文の「`else { B }`」の部分

▷ B が空の時は、「`else {}`」としても良いが、それを省略しても良い (`if (C) { A } else { B }`)==(`if (C) { A }`)

▷ then 節 (`{ A }` の所) は省略できない (`if (C) {} else { B }`)は(`if (C) else { B }`)とはできない

□ if 構文の「入れ子」

○ if 構文の文の中に if 構文をいれる

▷ 複数の条件の「条件の組み合わせ」を if 構文の組み合わせで実現できる

▷ 例 : `if ('A' <= ch) { if (ch <= 'Z') { /* ch が大文字の時 */ } else {} } else {}`

□ else if 構文 (イデオム)

○ else 節の文全体が if 構文(入れ子)になっている場合

▷ else 節のブレースを省略する

▷ 例: `if (C1) { A } else { if (C2) { B } else { C } }` => `if (C1) { A } else if (C2) { B } else { C }`

○ 入れ子による、段の深化が防げる

▷ 一つのデータに対して、複数の条件の検査になっている

▷ 「表現」と「意味」の対応 (イデオム)

推奨しない if 構文の表現

□ その他の表現

- いつでも、「基本」を守る事が肝要

- ▶自分で作成する時には、まずは、基本を守る

- 他人は、基本を守るとは限らないので、いろいろな記法を学ぶ事も必要

□ ブレースの省略

- then 節/else 節の文が一文の時にはブレースを省略できる

- ▶else if 構文は、その規則を特別な場合に適用している

- ブレース省略の是非

- ▶「省略」できるから、「手間」は減る (僅かな利点)

- ▶条件の適用範囲が不明瞭になる (大きな損失)

- ▶後から、変更を加える時に不便 (ミスの増大)

□ ぶら下がり構文の悲劇

- ブレースの省略と else の省略が重なると悲惨

- ▶if () { if () { else { } は、次のどちらか？

- if () { { if () { else { } }

- if () { { if () { } } else { }

switch 構文

ソフトウェア概論 A/B (2021/06/25)

switch 構文

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

switch 構文

□ switch - case 構文

○ 機能

▶ 多分岐構文 (if 構文は 2 分岐だが switch 構文は n 分岐)

○ 構文

▶ `<switch 文> ::= switch (<式>) { <case 並び> }`

▶ `<case 並び> ::= <case> の繰り返し`

▶ `<case> ::= <label> : 文並び`

▶ `<label> := case <定数> | default`

▶ ※1 一つの `<switch 文>` 内には、同じ `<label>` を含める事はできない

▶ ※2 文並び中に `break` 文があり、それが実行されると、`switch` 文は終了する

○ 意味

▶ もし、「式」の値が「定数」の何れかであれば、そのラベル以下の文を実行する

▶ そうでなくて、もし、`default` があれば、そのラベル以下の文を実行する

▶ そのいずれでもなければ、何もしない

おしまい

ソフトウェア概論 A/B (2021/06/25)

おしまい

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます