ソフトウェア概論 A/B

-- オセロゲーム板(2) --

数学科 栗野 俊一 / 渡辺 俊一

2013/10/25 ソフトウェア概論

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 (2013/10/18) の復習

- 前回 (2013/10/18) の復習
  - オセロゲームのゲーム盤の作成
    - ゲーム盤のデータ構造の設計
    - step by step プログラミング (少しずつ機能を確認しながらプログラムする)
  - C 言語の概念
    - シンボル定数(名前付きの定数)の定義と #define (マジックナンバー)
    - 二次元配列 (配列の配列)
    - for 文

お知らせ

- 本日の予定
  - オセロゲーム板を作る(そのニ)
- 本日の目標
  - 演習
    - オセロゲーム板を動かしてみる(そのニ)

前回 (2013/10/18) の課題

- 前回 (2013/10/18) の課題
  - なし

本日の課題 (2013/10/25)

- 本日 (2013/10/25) の課題
  - なし

当分の方針

- 暫く
  - オセロのゲーム盤を作ってみる

二次元配列

- 配列：同じ物が、一列に並んだ物 (一次元配列)
  - 配列の配列は？ (配列が二次元に並んでいる)：二次元配列となる
    - BInfo info[BOARD_SIZE][BOARD_SIZE]; info[i][j] が二次元に並んでいる

	0	1	..	6	7
0	info[0][0]	info[0][1]	..	info[0][6]	info[0][7]
1	info[1][0]	info[1][1]	..	info[1][6]	info[1][7]
			..		
6	info[6][0]	info[6][1]	..	info[6][6]	info[6][7]
7	info[7][0]	info[7][1]	..	info[7][6]	info[7][7]

  - k 次元の配列の配列は？
    - k+1 次元の配列になる

配列と文字列

- 文字列とは？
  - 文字の並びだった.. という事は..
    - char の一次元配列が「文字列」？
- 答は Yes でもあり No でもある
  - Yes :「文字列」は「char の一次元配列」で実現されている
    - char の一次元配列にできることは「文字列」にもできる
    - 「文字列」を「char の一次元配列」のように扱って良い
  - No : 逆は真ではない(char の一次元配列は、文字列とは限らない)
    - char の一次元の配列の要素を文字にし最後に EOS(\0)を入れれば文字列と同様に振る舞う
    - C 言語のコンパイラが、「文字列」を特別扱いしてくれる

分割コンパイル

- 分割コンパイル
  - 一つのファイルに全部入れても良いが..
    - 複数に分割する事も出来る
  - 分けるメリット
    - 柔軟性が生れる：リンク時にも制御できる(Test Case)
    - 相互の影響が小くなる：編集時の置換の失敗
  - 分けるデメリット
    - 情報の共有が難しい：ヘッダーファイルの利用
    - コンパイルリンクの手間手間が増える：make & Makefile の利用

色々な変数の宣言とスコープ

- 変数の宣言
  - これまでは関数の中だけだった(ローカル変数)
    - 関数の中だけで有効
    - ブロック内だけで有効な変数もつくれた
- 関数の外でも変数は宣言できる(グローバル変数)
  - 複数の関数から共通の変数を利用する事ができる
    - 関数間に変数を経由した「結び付き」ができる
  - グローバル変数の得失
    - 「結び付き」が「便利さを生む」のは事実(便利なので)
    - 常に意識する必要がある(腐れ縁になってしまう可能性が..)
  - グローバル変数の利用はできるだけさける
    - 情報は、引数と返り値でやり取りする
    - ただし、効率は悪くなる(コピーがおきるので..)
    - 効率を高める方法はあるが..(後に、ポインターの話をする)