

精密工学科プログラミング基礎

第1回資料 (10/9実施)

今回の授業で習得してほしいこと:

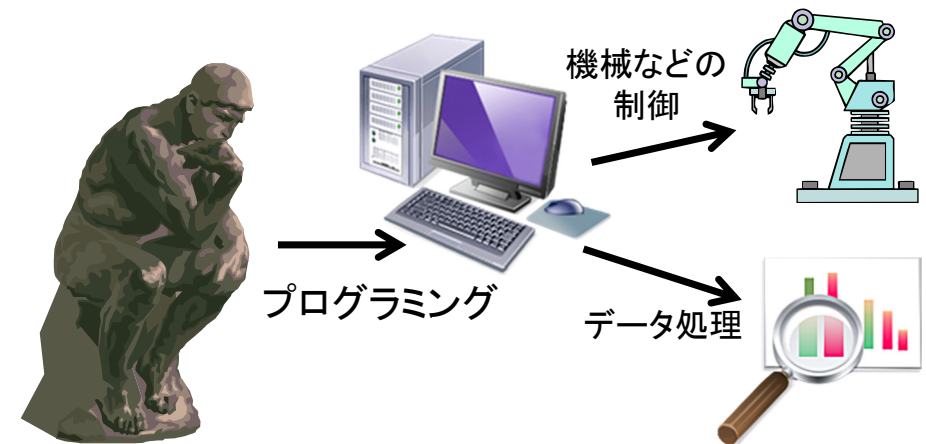
- C言語プログラムのコンパイルと実行
 - プログラムエディタによるソースコードの作成
 - コンパイラ “gcc” によるソースコードのコンパイル
 - コマンドラインからのプログラムの実行
- 計算結果を表示する方法
 - 関数 “printf” の使い方 (WEBで調べながら学ぶ)

資料のURL : <http://lecture.ecc.u-tokyo.ac.jp/~tohtake/>

1

プログラミングとは？

- 我々の意図する処理を行うように
計算機に指示を与える行為 [Wikipedia]



2

本授業で学んでほしいこと

- C言語の基礎
 - 現在よく使われている Java, C++, C# も、
C言語ができれば習得は容易
- C, C++, C#, BASIC, Java, Mathematica, Matlab, FORTRAN, LISP, Occam, Pascal, Prolog, Python, Ruby, ...
- プログラミングは単に「ツール」であること
 - できるのは計算だけ
 - 何を行うかを考えることが重要

3

本授業の対象

- プログラミング未経験者
- すでにC言語をマスターしている人は、
 - 自分で課題を作ってみる
 - 隣の人に教える
 - 追加の課題をやってみる (HP一番下を参照)

4

計算機室のパソコンで プログラミングする時の手順

1. student アカウントでログイン
2. プログラムを入れておくフォルダを作る
3. プログラムのソースコードを作成 (p.6-7)
4. コンパイルして実行する (p.8-10)
5. 授業が終わってないならば、手順3に戻る
6. 手順2で作ったフォルダをUSBメモリや
おススメ → Google drive などにバックアップする！
7. パソコンをシャットダウンする

5

ソースコード “hello.c” の作成

- ① ソースコードを置くフォルダを開いて、
右クリック→新規作成→テキストドキュメント
- ② 作成したテキストファイルの名前を “hello.c” にする
– 拡張子を .txt から .c に変える必要があります
- ③ ファイルをダブルクリックしてエディタを開く
- ④ 右のソースコードを打ち込む
- ⑤ 保存する

```
#include <stdio.h>
int main(void)
{
    printf("Hello.¥n");
}
```

6

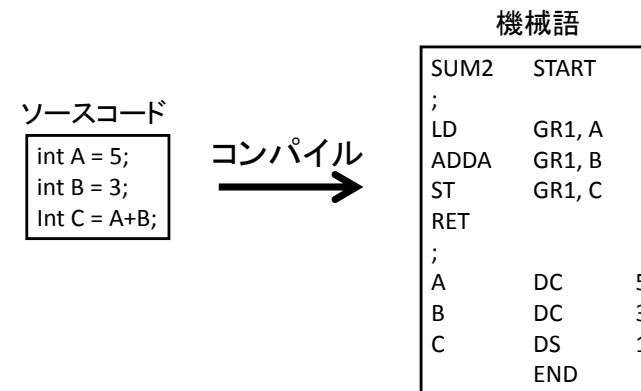
プログラムエディタについて

- マイクロソフト社 Visual Studio の
プログラムエディタを利用します
– Emacs が好きな人はそちらでもOKです
- 便利なショートカットキー
おススメ →
– Ctrl + S: 保存
– Ctrl + K、Ctrl + F: マウスで選択したコード行を
読みやすくする
– Ctrl + K、Ctrl + C: 選択した行をコメントアウトする
– Ctrl + K、Ctrl + U: 選択した行のコメントアウトを解除

7

プログラムのコンパイルとは？

- コンパイル：
プログラムが記述されたソースコードを、
計算機で実行可能な機械語に翻訳すること

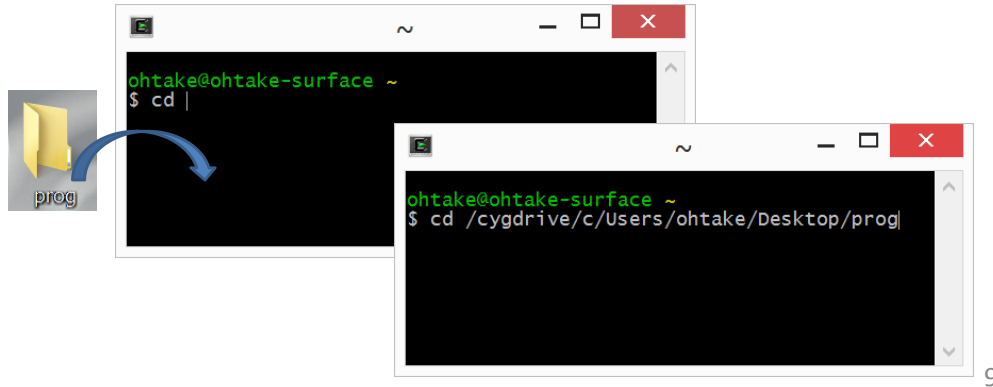


※ コンパイルが必要でないプログラミング言語をインタプリタ型言語という

8

コンパイルの準備

- ① スタートメニューから“Cygwin Terminal”を選択
- ② “cd”と打つ（Enterはまだ押さない）
- ③ 作ったフォルダーをドラッグ&ドロップ
- ④ Enterを押してフォルダの中に移動



コンパイルと実行

- ① ターミナルでソースコードをコンパイル
gcc hello.c ↵
- gcc がコンパイルのためのコマンド
- ② できた実行ファイル“a.exe”を実行する
./a.exe ↵
- “./”は、ここ(.)にある(/)という意味

おススメ

- ターミナルでは、ファイル名などを途中までタイプして、
Tab キーを押すと補間ができる
- ↑キーで、これまで実行したコマンドをさかのぼることができる

10

プログラミングを学ぶときのコツ (私見)

- とりあえず書いてみて、間違いを直していく
- 「デバック」という専用の言葉がある程、
間違いは日常茶飯事
- プログラムはどんな書き方でも
正しく動作すればよい
- エレガントな書き方を追及する必要はない
- WEB 検索を活用する
- 情報の良し悪しを見極めることが重要

11