プログラミング基礎 I

第 1 回

10/6

東京大学大学院工学系研究科 総合研究機構 / 精密工学専攻

禹 ハンウル

woo@robot.t.u-tokyo.ac.jp

プログラミング基礎 I

- 時間：A1ターム 水曜1限・2限
- 講義の目的：C言語によるプログラミングの基礎を学ぶ
- 講義の進め方：最初に内容を説明し，課題をやってもらう

  実習が中心，実習中は適宜休憩を取る

  課題が終わったら退室しても構わない
- 成績評価：レポート（期末），授業中の課題（毎回）
- 講義のホームページ：https://www.hanwoolwoo.com/lecture
- テキストのPW：[redacted]

講義の進め方

・Zoomの名前を「学籍番号_名字」にする

・講義中の質問はチャットに書き込む

・実習中に質問がある場合は，「質問があります」と下記のサイトに書き込む

・課題のチェックをお願いする場合は，「課題チェックをお願いします」と書く

・教員がリストの順番にブレークアウトルームへ移動させる

・ブレークアウトルームに入ったら，TAに質問（課題チェック）する

質問シートのリンク

https://docs.google.com/spreadsheets/d/1E1wLvEuibjac0Hr6fdmt9fEOa2YSG4NEn5LMD7Jy_rU/edit?usp=sharing

プログラミングの環境構築

本講義は，自分のパソコンでプログラミングを行う実習形式となります.

プログラミングを行うためには，「環境構築」が必要です.

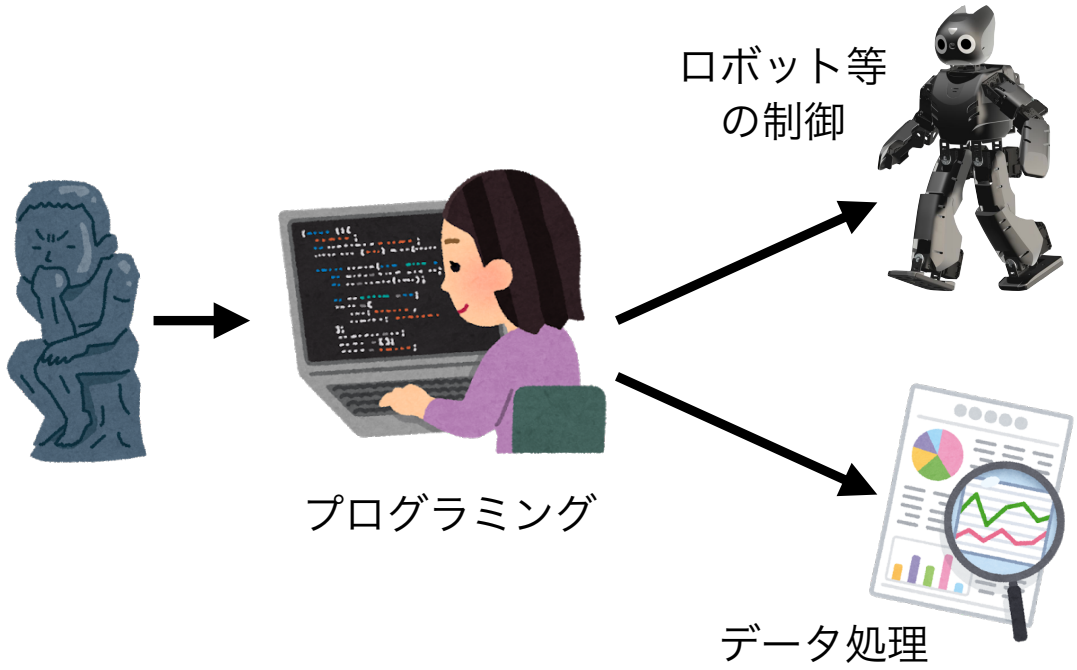
下記のサイトを参考にし，各自行ってください！

https://www.hanwoolwoo.com/lecture

プログラミングとは？

[Wikipedia]

人間の意図した処理を行うように，コンピュータに指示を与える行為



本講義で学んでほしいこと

C言語の基礎

様々なプログラミング言語が存在するが，C言語の基礎ができていれば，他の言語も取得が用意

他の言語の例：Python, Java, C++ C#, Ruby

アルゴリズムの実装

プログラミング言語は単なるツールであり，自分のアイディアを実装する力が重要

本講義の対象

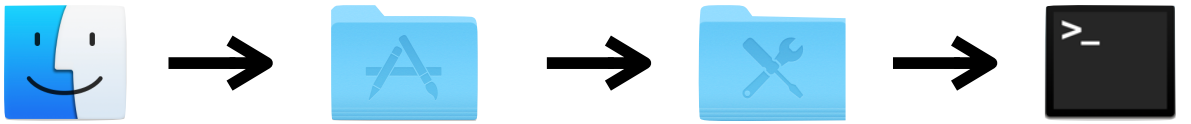
プログラミング初心者

講義のスケジュール

#	授業日時		授業内容	備考
1	10/6(水)	1限	C言語プログラムの作成・コンパイル・実行，演習【TA】	
		2限		
2	10/13(水)	1限	計算機上での数値の表し方，プログラムにおける変数、演習【TA】	
		2限		
3	10/20(水)	1限	繰り返し，演習【TA】	
		2限		
4	10/27(水)	1限	条件分岐，演習【TA】	
		2限		
5	11/10(水)	1限	関数1，数値解析法，演習【TA】	
		2限		
6	11/17(水)	1限	レポート課題(自習)	
		2限		
7	11/24(水)	1限	授業無し (後日、レポート課題提出)	
		2限		

本講義用ディレクトリ（フォルダ）作成

1．ターミナルを起動



2．任意の場所にディレクトリを作成

- cd コマンドでデスクトップに移動

```
$ cd Desktop
```

- mkdir コマンドでディレクトリ prog を作成

```
$ mkdir prog
```

- cd コマンドで，作成したディレクトリへ移動

```
$ cd prog
```

ソースコードの作成

1．hello.c という名前のファイルを作成

```
$ touch hello.c
```

2．atom（他のエディタでも良い）でファイルを開く

```
$ open -a Atom.app hello.c
```

3．以下のソースコードを入力

```c
#include <stdio.h>

int main(void)
{
  printf("Hello.\n");
  return 0;
}
```

注意！！
PDFのコードをコピペしても動かない！！
必ず手入力でお願いします！

4．保存する（commandキー + s）

Tips

Macでバックスラッシュを出す

- システム環境 → キーボード → 入力ソース → 日本
- “￥”で入力する文字を“\”に変更

コンパイルと実行

コンパイル

人間が記述したコードを計算機で実行可能な機械語に翻訳すること.
本講義では，gccというコンパイラを使ってソースコードをコンパイルする.

```
$ gcc -o hello.out hello.c
```

正しくコンパイルできると，hello.out という名前の実行ファイルが生成される

実行

```
$ ./hello.out
```

ファイル構成

C言語プログラムは基本的に，
.c ファイル：ソースファイル
.h ファイル：ヘッダファイル
の２種類のファイルから構成される

```c
#include <stdio.h>

int main(void)
{
  printf("Hello.\n");
  return 0;
}
```

ヘッダファイルは，他のファイルに実装された機能を使うためのもの．
これをインクルードすることにより，その機能を使うことができるようになる．

例）開発済みのライブラリを使う，複数の開発者が分担して開発を行う

ソースコードの解説

```
#include <stdio.h> /* 標準入出力（Standard I/O）に関するヘッダファイルをインクルードする */

int main(void) /* main関数 */
{
    printf(“Hello.\n”); /* printf関数を使って文字列を画面に表示する */

    return 0; /* 関数の実行が終わった時に0という値を返す */
}
```

main関数

・C言語プログラムは関数（function）によって定義される.

（関数については，第５回の講義で詳細に説明する）

・実行プログラムには，必ず main関数が存在する.

・１つのアプリケーションに，main関数は１つしか存在しない.

・プログラムは，main関数から実行される.

文とブロック

関数の本体は,
{文　文　文　…}のように，{}の中に計算などを行う文が並ぶ.

```c
#include <stdio.h>

int main(void)
{
  printf("Hello.\n");
  return 0;
}
```

文は；（セミコロン）で区切られる.
複数の文を{}で囲ったものをブロックという.

文は，基本的に並んでいる順に実行される.

フォーマット

C言語はフリーフォーマットなので，以下のように書くことも可能.

```c
#include <stdio.h>
int main(void){printf("Hello.\n");return 0;}
```

ただし，可読性が下がるので，通常は改行とタブを使って整列させる！

：ソフトウェアの品質に大きく影響する

：コーディング規約

数字の出力

number.c

```c
#include <stdio.h>

int main(void)
{
  int i = 10;
  double j = 5.4583;

  printf("i is %d, j is %f\n", i, j);

  return 0;
}
```

プログラミングを学ぶときのコツ

１．Web検索を適切に活用する

ただし，情報の良し悪しを見極める必要がある.

拾ってきたコードにエラーが含まれていると，発見するのが極めて難しい.

２．自分で手入力する

コピペするだけだと，中身を理解できない.

時間がかかっても初期は自分の手で入力することをお勧めする.

課題１

テキスト５ページまでのコードをエディタで入力し，コンパイルして実行する

- hello.c（図１）
- hello2.c（図２）
- fahren.c（図４）

注意！！
PDFのコードをコピペしても動かない！！
必ず手入力でお願いします！

hello.c

```c
#include <stdio.h>

int main(void)
{
  printf("Hello.\n");
  return 0;
}
```

hello2.c

```c
#include <stdio.h>

int main(){ printf("Hello2.\n");}

```

fahren.c

```c
#include <stdio.h>

void main(void) /* main function  */
{
  printf("0 F is %7.1f C\n", 5.0/9.0*(-32)); /* コメント */
  printf("50 F is %7.1f C\n", 5.0/9.0*(50-32));
  printf("100 F is %7.1f C\n", 5.0/9.0*(100-32));
}
```

課題２

fahren.cの3行のprintf文に対して，各行の意味がわかるようにコメントを入れる

：/＊ コメント内容 ＊/

```c
#include <stdio.h>

void main(void) /* main function  */
{
  printf("0 F is %7.1f C\n", 5.0/9.0*(-32)); /* コメント */
  printf("50 F is %7.1f C\n", 5.0/9.0*(50-32));
  printf("100 F is %7.1f C\n", 5.0/9.0*(100-32));
}
```

課題３

fahren.cで３行にわたって表示される数値を小数点３桁までにそろえて表示されるようにしてください.

：printfに関連する情報を探してみてください.

：キーワードは「精度（小数点何桁まで表示するか）」です.