

コンピュータ実習
第3回 2023年10月19日

C言語の基礎

数値計算

入出力プログラム

条件判断とループ

担当: 相澤 直人, 宍戸 博紀

コンピュータ実習(第3回)

C言語の基礎

プログラムとは？

プログラムの作法

デバッグ

例題：

数値計算

データの入出力(端末から・ファイルから)

条件判断・ループ処理

コンピュータ実習(第3回)

まずはプログラムソースを作成します

補助教材

コンパイルから実行までの流れを開いてください

エディタを立ち上げてください

サンプルプログラムの作成のところにある
`test1.c`を入力し、保存してください

コンピュータ実習(第3回)

ソースファイルのコンパイル

では, そのソースファイルを実行できるようにします. この作業を**コンパイルする**といいます.

ターミナルを立ち上げて, 補助教材にあるとおり

```
cc -o test1 test1.c
```

とターミナルから入力してください.

コンピュータ実習(第3回)

プログラムの実行

コンパイルが終了したらコンパイルを実行したターミナルから

```
./test1
```

と入力して実行してみてください。

コンピュータ実習(第3回)

まずはプログラムソースを作成します

今cのソースファイルと呼ばれる、いわゆるプログラムを入力し、コンパイルと実行をおこないました

では、ソースの中身を解説します。良く聞いてください。

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

```
/* Program "test1.c" */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("Practice C Programming!! ¥n");
```

```
    return 0;
```

```
}
```

コンピュータ実習(第3回)

プログラムから実行までの流れ

① “.c” という拡張子のテキストファイルにC言語のプログラムを記述する

テキストエディタに記述したプログラムを**ソースプログラム**, そのファイルを**ソースファイル**といいます.

② ソースファイルをコンパイルする

コンパイルしてできたファイルを**オブジェクトファイル**といいます.

③ オブジェクトファイルをリンクする

リンクとはプログラム実行に必要なファイルを結合することです. リンクが成功すると実行可能ファイルができます.

コンピュータ実習(第3回)

プログラム記述時の約束

- ①原則として半角で記述する
- ②半角カナは使わない
- ③全角スペースの使用に注意する
- ④小文字と大文字を区別して書く
たとえば, ifとIFはまったく別物です.
- ⑤コメントは/*と*/でくる
プログラムに反映したくない説明的な記述を/* */の中に書くことができます.
- ⑥予約語に気を付ける
予約語はコンパイラが使用するキーワードです. それぞれの持つ働き以外の目的で使用することはできません. (例: auto, const, enum)

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

プログラムは

関数・変数・定数・演算の集まり と考えてください。

では、関数とは何か

実行すると何らかの結果を出すもの

例えば、

計算して答えを返してくれる

データを受け取るだけ

与えられた文字などを表示させる

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

関数はかならずどういう動作をするかを定義しなければなりません.

関数は,

関数には自分で作る(定義する)もの

と

最初から用意(定義)されているもの

(標準ライブラリ関数)

に分かれます.

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

```
/* Program "test1.c" */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("Practice C Programming!! ¥n");
```

```
    return 0;
```

```
}
```

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

```
/* Program "test1.c" */
```

←コメント

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
printf("Practice C Programming!! \n");
```

```
return 0;
```

```
}
```

main関数, 特別な関数です.

←

関数の一般的な形は

hogehoge(引数)

その関数の内容はreturnで終わり,

{ }で囲まれます

※return 0は正常終了を意味します.

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

```
/* Program "test1.c" */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("Practice C Programming!\n");
```

```
    return 0;
```

```
}
```

int main(void)

- ・内容を順次実行していく幹の部分
- ・通常は引数を渡さない

・ただし、コンパイル後の実行ファイルの実行時に引数を渡したいときは定義する場合もある

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

```
/* Program "test1.c" */
```

printf関数

```
#include <stdio.h>
```

()内の引数を表示しろ という関数

ここで注意. この関数の定義はどうする?

```
int main(void)
```

```
{
```

```
    printf("Practice C Programming!! ¥n");
```

```
    return 0;
```

```
}
```

コンピュータ実習(第3回)

ソースファイル(プログラム)の構造

```
/* Program "test1.c" */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("Practice C Program");
```

```
    return 0;
```

```
}
```

だれもが使うような関数のソースファイルはあらかじめ用意されている。(ヘッダファイル)

今回は標準的な入出力用の関数を定義したソースファイルを読み込んでいる。