

## プリプロセッサ

プリプロセッサとは、コンパイル（翻訳）に先だってファイルの取り込みや文字列の置き換えなどの前処理を行うものである。そこで使用される代表的な制御文には、

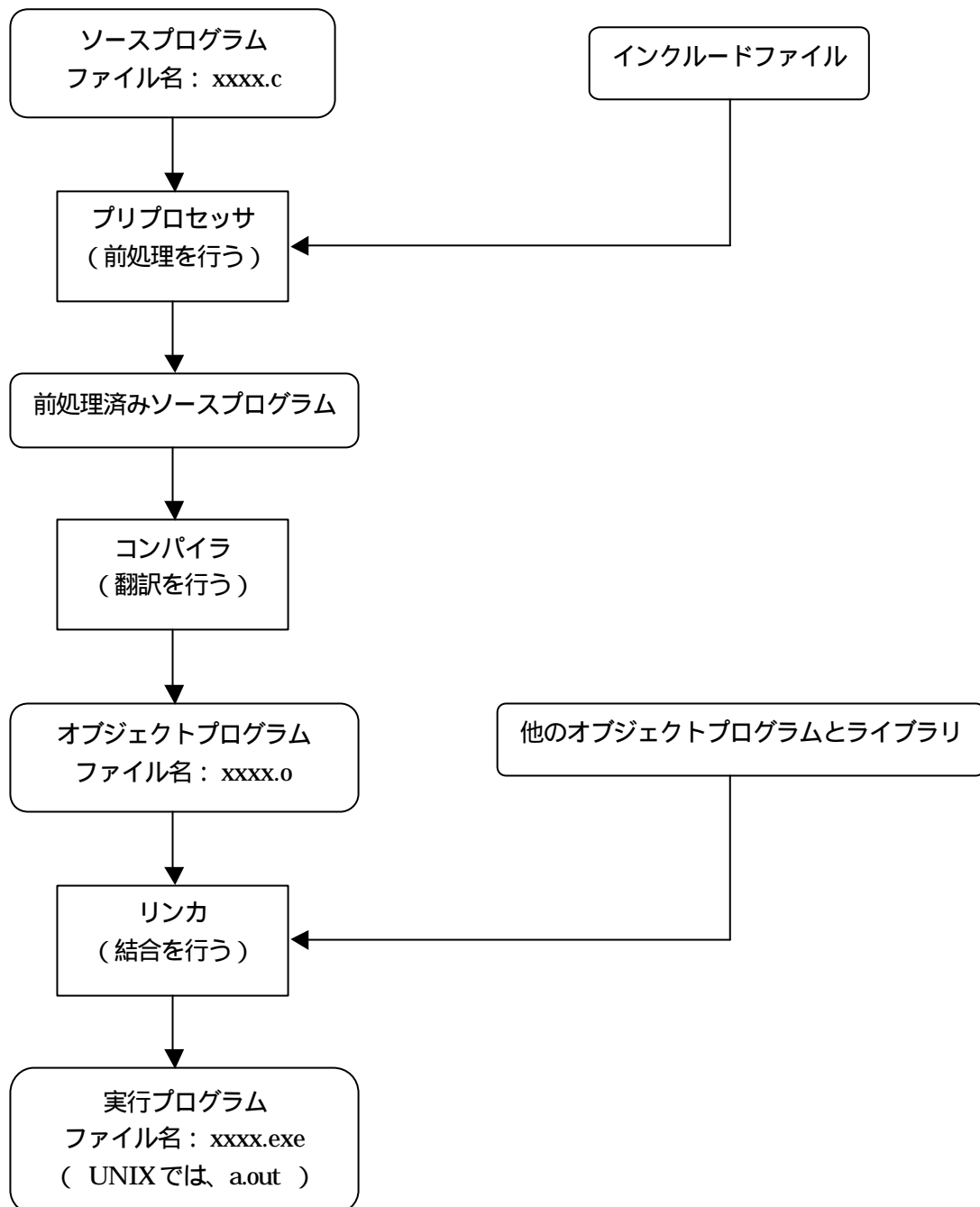
```
#include ..... ファイルの取り込み（取り込まれるファイルをインクルードファイルと呼ぶ）
#define ..... 文字列の置き換え
```

がある。

C言語においては、厳密にいうと

1. プリプロセス（前処理）
2. コンパイル（翻訳）
3. リンク（結合）

という3段階の処理を経て実行プログラムが作成される。Cでソースプログラムから実行プログラムを得るまでの流れは以下のとおり。



前処理のための制御文の使用例)

```
#include <stdio.h>          /* システムに標準的に備わっているヘッダファイルの取り込み */
#include <math.h>           /* 同 上 */
#include "my_own.h"        /* 自作のヘッダファイルの取り込み */

#define PAI      3.141592   /* マクロ：文字列の置き換え */
#define add(x,y) ((x)+(y)) /* 引数付きマクロ */

void main(void)
{
    double  xdeg=45.0, ydeg=15.0, xrad;

    xrad= add(xdeg, ydeg)*PAI/180.0;
    PRT("angle: %f[degree] %f[radian] cos: %f\n", xdeg+ydeg, xrad, cos(xrad) );
}
```

-----自作ヘッダファイルmy\_own.h の内容-----

```
#define PRT printf
-----おしまい-----

-----実行開始-----
angle: 60.000000[degree] 1.047197[radian] cos: 0.500000
-----おしまい-----
```

## 1. #include 文

この制御文の置かれた所に、ファイルを取り込む(挿入する)ことを意味する。指定されるファイルを総括してインクルードファイルと呼ぶが、特に ".h" という名前のついているファイルをヘッダファイルと呼ぶ。

C言語処理システムに標準的に用意されているヘッダファイル(stdio.hなど)を指定する場合には、

```
#include <ファイル名>
```

のように、< > でファイル名を括る。

- ・stdio.h には、標準入出力関数 ( printf や scanf など) についてのプロトタイプ宣言が記述されている。
- ・math.h には、数学関数についてのプロトタイプ宣言が記述されている。

ユーザが自分で作成したヘッダファイルを指定する場合には、

```
#include "ファイル名"
```

のように、" " でファイル名を括る。

## 2. #define 文

### 2.1 マクロの定義

#define 文とは、文字列の置き換えを指示する制御文である。置き換えられる文字列をマクロと呼び、

```
#define マクロ 置き換える文字列
```

により定義する。上の使用例では、

```
#define PAI      3.141592
```

として、プログラム中に PAI がでてきたらそれを 3.141592 に置き換えるよう指示している。  
 ここでのマクロ PAI は記号定数と呼ばれ、記号定数は通常大文字で記述される（小文字でも誤りではない）、  
 また関数名を置き換える例として、自作のヘッダーファイル my\_own.h のなかで、

```
#define PRT printf
```

により、PRT を printf という文字列に置き換える指示をしている。その結果、プログラム内で

```
PRT("angle: %f[degree] %f[radian] cos: %f\n", xdeg+ydeg, xrad, cos(xrad) );
```

のように PRT と記述されると、printf 関数が呼ばれることになる。

## 2.2 引数付きマクロの定義

使用例

```
#define add(x,y) ((x)+(y))
```

のように、マクロでは引数をとることができる。例えばプログラム内に

```
xrad= add(xdeg, ydeg)*PAI/180.0;
```

と記述されている時、プリプロセッサで文字列の置き換えを処理すると、

```
xrad= ((xdeg)+(ydeg))*PAI/180.0;
```

となる。

引数付きマクロを定義する時、置き換える側の各引数と文字列全体を括弧()でくくることが推奨される。これは誤動作を防止するためである。

例えば上の例

```
#define add(x,y) ((x)+(y))
```

で括弧を省いて

```
#define add(x,y) x+y
```

と定義してみると、これはプリプロセッサで処理した後、

```
xrad= xdeg+ydeg*PAI/180.0;
```

となってしまう、演算順序が異なってくる。

つまり、+ - の演算よりも \* / の演算のほうが優先順位が高いため、

```
xrad= xdeg+(ydeg*PAI/180.0);
```

と同じことになってしまう。

## 課題（関数）

教科書の例3 - 2 . 2 に倣って、10 個の整数データをコンソールから入力すると、平均値、最大値、最小値を出力するプログラムを作成せよ。

データ入力、平均値の算出、最大値の検出、最小値の検出はそれぞれ関数にすること。

提出期限： 11月29日（木）

提出物： 課題のプログラム

\* プログラムの第1行目には、コメントで（ /\* \*/ でくくって）学籍番号と名前を記述すること

\* プログラムファイル名は p○○××e5.c とすること

○○は学籍番号の上2桁の数字

××は学籍番号の下2桁の数字

提出形態： E-mail に上記提出ファイルを添付すること

- ・あて先 [cp2-69@cc.tuat.ac.jp](mailto:cp2-69@cc.tuat.ac.jp)
- ・題 cp2ex5
- ・本文 感想、学籍番号、名前を記述