

C言語を用いたプログラミング8

Key Words:

- ・ファイルからのデータ入力
- ・ファイルへのデータ出力
- ・ファイルポインタ
- ・ファイル入出力関数
 - ・fopen, fclose
 - ・fscanf, fprintf
- ・CSVファイル

プログラム



- ・データを記憶するための変数(データ構造)
 - ・整数型
 - ・実数型
 - ・文字型
 複数のデータを記憶 → 配列 (数値リスト, ベクトル, 行列, ...)
- ・処理をするための制御構造(アルゴリズム)
 - ・順次
 - ・選択
 - ・反復
 モジュール化 → 関数

ファイル入出力の目的

(処理例) 次の中から最大値を求めなさい。

39.1	98.3	-44.6	59.0	32.5	38.3	48.9	-31.6
29.5	61.2	40.3	-31.9	87.5	35.8	-61.3	37.9
52.1	-85.9	29.0	-69.3	67.3	62.7	-68.3	48.1

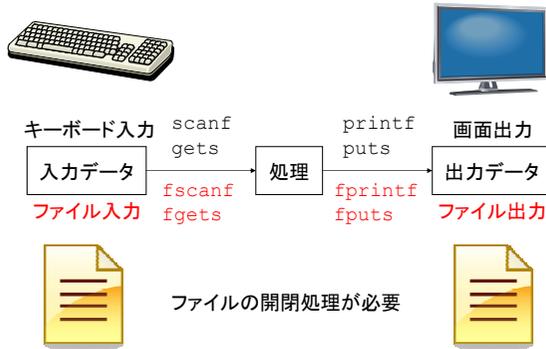
キーボード入力の短所

- ・多数のデータを入力するのは面倒
- ・入力を間違えることがある

ファイルを利用する長所

- ・膨大なデータを容易に処理できる。
- ・誤データを簡単に修正できる。
- ・データをファイルに記録しておけば、様々な処理に利用できる。
 気象観測データ: 膨大なデータ
 衝突試験データ: 瞬間のデータ

標準入出力 (stdio.h)



ファイル開閉の書式

```
FILE *fp;
fp=fopen("ファイル名", "モード");

// ファイル・オープンのエラー処理

// ファイル・データの入出力処理

fclose(fp);
```

ファイル情報用の変数
ファイルを開く

ファイルを閉じる

指定	ファイル名(例)	モード	意味
相対指定	abc.txt	r	読み込み (read)
絶対指定	Z:¥¥it2c¥¥abc.txt	w	書出し (write)
		a	追加 (append)

- ・相対指定は、プログラム実行時のフォルダが基点
- ・「¥」は機能コードのため、「¥¥」と重ねて記述する
- ・モードには、他にr+, w+, a+, rb, wb, abなどがある

ファイル入出力関数 (stdio.h)

関数	機能	書式	エラー時の戻り値
fopen	開く	fopen("ファイル名", "モード")	NULL
fclose	閉じる	fclose(ファイルポインタ)	
fscanf	入力	fscanf(ファイルポインタ, ...)	EOF
fprintf	出力	fprintf(ファイルポインタ, ...)	

- 関数fscanf, fprintfの書式は、ファイルポインタを第1引数に追加する以外は関数scanf, printfと同じ
- fscanfは空白および改行をデータの区切りとして認識する
- fscanfはファイル終端でEOFを出力する
- FILE, NULL, EOFは太文字で指定する。FILEは型、NULL, EOFは数値定数としてstdio.hに定義されている

ファイル入出力関数の仕様 (stdio.h)

機能	書式	成功時	失敗時
ファイルを開く	FILE *fopen(char *fname, char *mode)	ファイルポインタ (ファイル情報の格納場所)	NULL
ファイルを閉じる	int fclose(FILE *fp)	0	EOF
ファイル入力	int fscanf(FILE *fp, char *fmt, ...)	代入に成功したフィールド数	EOF
ファイル出力	int fprintf(FILE *fp, char *fmt, ...)	出力バイト数	負の値

ファイル・オープン時のエラー処理

ファイルがオープンできない場合の対策が必要

```
関数の終了:
if (fp == NULL) {
    エラー処理
    return 終了コード;
}

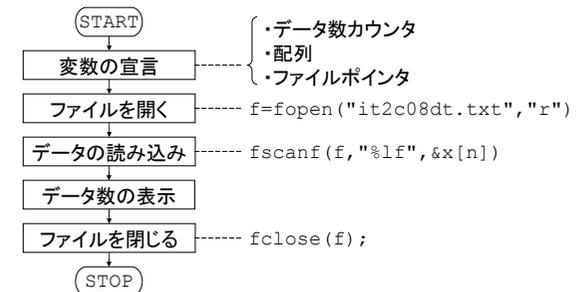
プログラムの終了:
if (fp == NULL) {
    エラー処理
    exit(終了コード);
}
```

関数	機能	書式	定義
exit	プログラムの終了	void exit(int status)	stdlib.h

```
stdio.h
#define EOF (-1) End Of File
#define NULL 0 無効の、値のない、零の
```

独自問題8.1

ファイル「it2c08dt.txt」に記録されたデータを配列に読み込むプログラムを作成しなさい。
 ・データおよびデータ数を表示しなさい。
 ・データファイルはmoodleに登録されている。



独自問題8.1(1/2)

10

```
#include <stdio.h>

int main(void) {
    int n;
    double x[100];
    FILE *f;

    f = fopen("it2c08dt.txt", "r");
    if ( f == NULL ) {
        printf("ファイルが開けません\n");
        return 1;
    }
}
```

stdio.hを
必要とする命令

```
FILE
fopen()
fclose()
fscanf()
printf()
```

独自問題8.1(2/2)

11

```
n=0;
while ( fscanf(f,"%lf",&x[n]) != EOF ) {
    printf("x[%d]=%.1f\n",n,x[n]);
    n++;
}
printf("データ数=%d\n",n);

fclose(f);

return 0;
}
```

独自問題8.2

12

独自問題8.1のプログラムに次の処理を追加しない。
・関数sumを用いて合計を求めない。
・関数maxを用いて最大値を求めない。

上記の関数は前回作成した。貼り付けて利用しない。
(既存モジュールの再利用)

```
printf("合計=%f\n",sum(x,n));
printf("最大値=%f\n",max(x,n));
```

独自問題8.3

13

独自問題4.1の結果をファイル「20?????.csv」に出力しない

0~xf[rad]の区間において導関数値と差分値の計4種の値を刻み幅h[rad]で出力するプログラムを作成しない。区間xfと刻み幅hはキーボードより入力させない。

前方差分	$\frac{f(x+h)-f(x)}{h}$
後方差分	$\frac{f(x)-f(x-h)}{h}$
中央差分	$\frac{f(x+h)-f(x-h)}{2h}$

入力 (キーボード) 区間xf[rad]=6.2
刻み幅h[rad]=0.1 $f(x) = \sin x$

出力(ファイル「20?????.csv」) 学籍番号8桁+3

角度, 導関数, 前方差分, 後方差分, 中央差分
0.000000, 1.000000, 0.998334, 0.998334, 0.998334
0.100000, 0.995004, 0.988359, 0.998334, 0.993347
.....

独自問題8.3(1/2)

14

CSV (Comma Separated Value)
・データをコマンドで区切ったテキスト形式のファイル
・ファイルの拡張子を「csv」とする

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

int main(void) {
    double x,h,xf;
    FILE *f;

    f = fopen("19113???.csv", "w");
    if ( f == NULL ) {
        printf("ファイルが開けません\n");
        return 1;
    }
}
```

stdio.hを
必要とする命令

```
FILE
fopen()
fclose()
fprintf()
printf()
scanf()

math.hを  
必要とする命令
```

```
sin()
cos()
```

独自問題8.3(2/2)

15

```
printf("区間xf[rad]=");
scanf("%lf",&xf);
printf("刻み幅h[rad]=");
scanf("%lf",&h);

fprintf(f,"角度, 導関数, 前方差分, 後方差分, 中央差分\n");
for (x=0; x<=xf; x+=h) {
    fprintf(f,"%f,",x);
    fprintf(f,"%f,",cos(x));
    fprintf(f,"%f,", (sin(x+h)-sin(x))/h);
    fprintf(f,"%f,", (sin(x)-sin(x-h))/h);
    fprintf(f,"%f\n", (sin(x+h)-sin(x-h))/(2*h));
}

fclose(f);
return 0;
}
```

独自問題8.4

16

独自問題5.1の結果をファイル「20?????.csv」に出力しない

次の漸化式をi=1000まで計算し、ファイルに出力しない。
ただし、時間刻みはh=0.01[s]とする。

$$x_i = px_{i-1} + qx_{i-2}, \quad (i \geq 2)$$

$$\begin{cases} x_0 \\ x_1 = x_0 + v_0 h \\ p = (2m - bh)/m \\ q = -(m - bh + kh^2)/m \end{cases}$$

入力データ (キーボード) m=????
b=????
.....

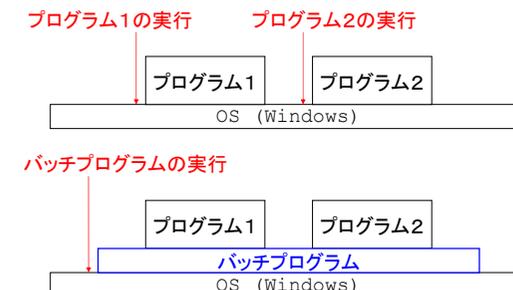
出力データ (ファイル) 時間, 位置
0.00, ...
0.01, ...
.....

入力データ (キーボード入力)	機械特性:m, b, k 初期値:x0, v0
出力データ (ファイル出力)	時間ti 位置xi

FAQ (main関数の戻り値)

17

Q 「return 1;」や「return 9;」など正常終了でない場合の数字は0以外なら何でも良いですか？
A はい、何でも良いです。
この値はバッチ処理と呼ばれるプログラムで利用できます。



プログラム

18

