
機能分子・生命化学科（機能分子工学科）
プログラミング演習

0 はじめに

0.1 授業 web サイト

<http://www1.doshisha.ac.jp/~bukka/lecture/programming/index.html>

0.2 2017 年度 授業計画

第 1 回	9/29	講義	第 0 章 はじめに 第 1 章 C プログラミングの基礎 第 2 章 入出力と演算子	
第 2 回	10/6	実習	仮想 Linux の使い方と代入に関する練習	[課題 1]
第 3 回	10/13	講義	第 3 章 処理の流れ（分岐）	
第 4 回	10/20	講義	第 3 章 処理の流れ（繰り返し）	
第 5 回	10/27	ここまでの復習	小テスト 1	
第 6 回	11/10	実習	第 3 章 処理の流れ（繰り返し）演習	[課題 2]
第 7 回	11/17	実習	第 3 章 処理の流れ（分岐）演習	[課題 3]
第 8 回	11/24	実習	第 3 章 分岐・繰り返し 演習	[課題 4]
第 9 回	12/1	ここまでの復習	小テスト 2	
第 10 回	12/8	講義	第 4 章 関数	
第 11 回	12/15	実習	数値積分の演習	[課題 5]
第 12 回	12/22	実習	非線形方程式の解法	[課題 6]
第 13 回	1/12	実習	配列・ファイルの入出力を利用したプログラム	[課題 7]
第 14 回	1/19	実習	複合的なプログラム	[課題 8]
第 15 回	1/29	最終確認	小テスト 3	

章の番号は教科書のもの

0.3 クラス分け

クラス名	対象者	実習担当者（及び TA）	実習教室	講義教室
A	第 6 回以降	木村佳文、熊崎茂一、中農浩史で分担	JM-402	TC3-104
B	に決定	(岡副眞也、川口匠悟)	JM-403	(合同)

講義担当者は日替わり

0.4 教科書

後藤良和，高田大二，中島寛和，『入門 C 言語』，実教出版

講義は教科書に沿って行うので，必ず購入すること

0.5 諸注意

- パソコン (Windows) のごく基本的な使用法は習得していることを前提とする。
- 仮想 Linux 上でのプログラミングを行う。使い方については第 2 回の授業で説明する。
- プログラムを保存する USB メモリを毎回持参すること。

- 遅刻厳禁。実習授業なので遅刻者のために説明を繰り返すことは不可能である。
- 受講生のクラス分けは 1 回目の小テスト終了後におこなう。
- 小テストは 3 回行う。初めの 2 回は、30-60 分程度の試験。試験後に解説を行う。最終の小テストは 60-90 分程度。初めの 2 回は 15 点、最終のテストは 20 点の配点である。
- 成績評価の方法についてはシラバスに記述してあるが、出席 (18 点) および提出課題 (4 点× 8=32 点) の評価で 50 点、小テストの合計 50 点でおこなう。

0.6 レポート提出について

- 課題は次の 2 つの両方を提出すること。
 - プリントアウト (出席チェックを兼ねる)

課題プログラムのソースファイルを印刷し、学生 ID と氏名を肉筆で記入して、各クラスの TA に提出すること。提出期限は各実習授業の終了時。最後までできていない場合も途中経過をかならず提出すること。出席チェックを兼ねるので、期限後の提出は一切認めない。また、完成後再提出の必要はない。
 - 電子ファイル

課題で作成したプログラムのソースファイルと実行結果とを、指示に従って一つのファイルにして、DUET のメッセージに対する回答の添付ファイルとして送信する。送信期限は実習授業翌週の火曜日の 24:00 とする。DUET のメッセージには 1 回しか回答できない。つまり、間違いに気がついていても再提出できないので、提出前に内容を十分確認すること。
- 電子ファイルの形式

ファイル名は `xxxxxxxx-XX.cpp` または `xxxxxxxx-XX.c` とすること。ただし `xxxxxxxx` は各自の学生 ID, `XX` は課題番号。

C 言語のソースプログラムの形式で、冒頭に注釈 (`/*` と `*/` ではさむ) として課題番号。次に同じく注釈として、学生 ID と氏名。次にプログラム本体。最後に注釈として実行結果。

```

/* プログラミング演習 課題 XX(課題番号) */
/* xxxxxxxx(各自の学生 ID) 各自の氏名 */
#include <stdio.h>

int main(void)
{
    ここには、プログラムの本体
}

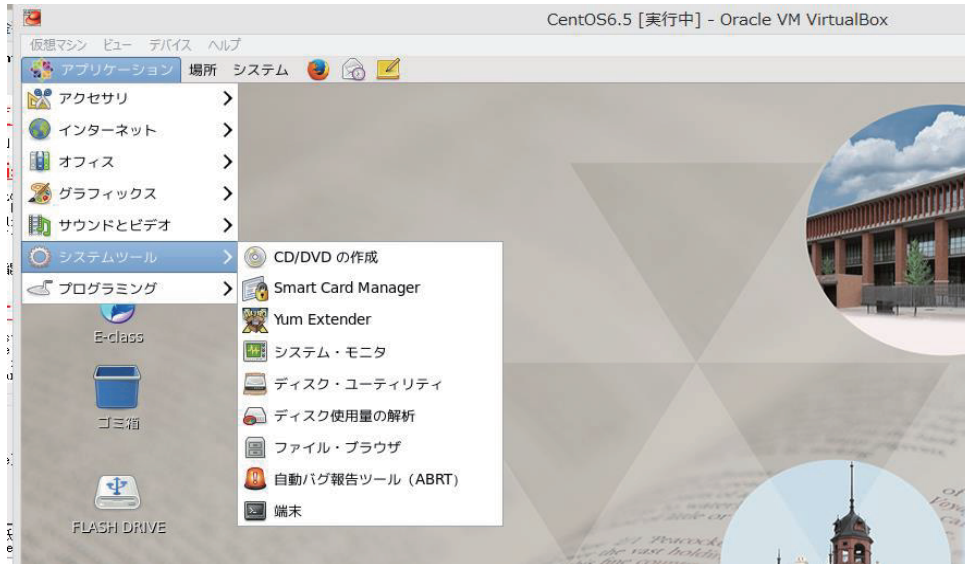
/*
実行結果
. . . . .
*/

```

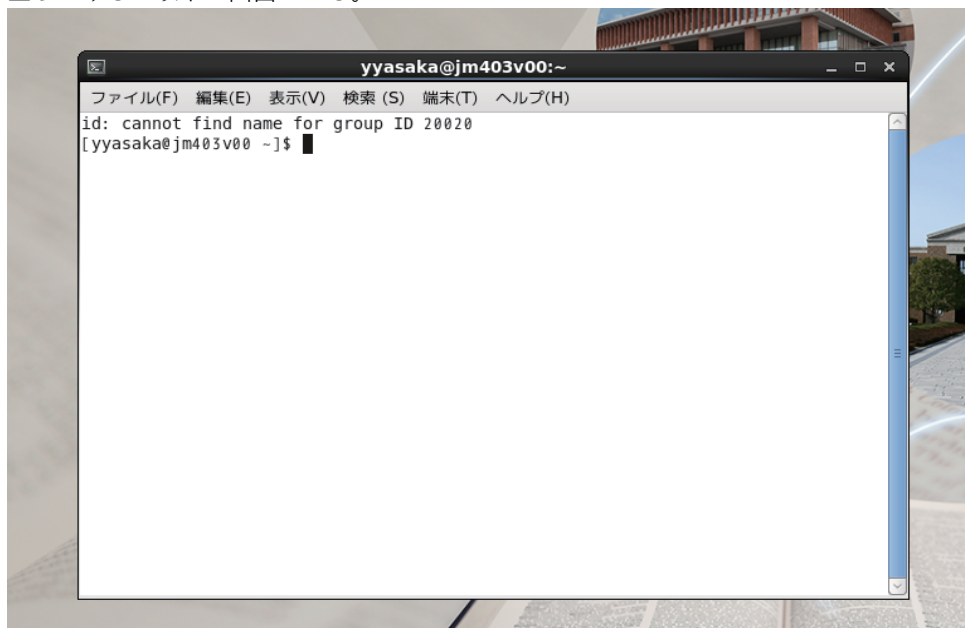
1 仮想 Linuxn の使い方

1.1 ターミナル

ターミナルの立ち上げ方は以下の画像のとおり。



立ち上げると以下の画面がでる。



1.1.1 ターミナルのコマンド

gedit	gedit を起動する。
cd .. (注 d と. の間にスペース必要、二つの・の間にはスペースを入れない)	ひとつ上のディレクトリに移動する。
cd <i>dirname</i> 例 cd home	配下の指定したディレクトリに移動する。 <i>dirname</i> にはディレクトリ名を代入する。
ls	カレントディレクトリの下にあるファイルのリストを表示する。
pwd	現在地 (カレントディレクトリ名) を表示する。
gcc -lm <i>progrname</i> -o <i>exename</i> 例 gcc -lm Hello.c -o Hello (注 gcc と- の間にスペース必要、- と lm の間はスペースを入れない)	プログラムのコンパイルをする。 <i>progrname</i> と <i>exename</i> には、それぞれソースプログラムファイル名と実行ファイル名を代入。
./ <i>exename</i> 例 ./Hello	プログラムを実行する。 <i>exename</i> には実行ファイル名を代入。
./ <i>exename</i> > <i>outfile</i> 例 ./Hello > Hello.txt	プログラムを実行した結果をファイルに保存する。 <i>exename</i> には実行ファイル名を代入、 <i>outfile</i> には任意のファイル名を代入。ただし、ファイル名の終わりに拡張子.txt をつけること。
上矢印キー)、(下矢印キー)	以前に実行したコマンドを表示させる。
rm <i>filename</i>	ファイルを消去する。 <i>filename</i> には消去したいファイル名を代入 注意： 一度消したファイルは回復できない。 rm * コマンドですべてのファイルが消去される
exit	ターミナルを終了する。

1.1.2 gedit の操作、設定

1. ファイルを保存する。ファイル名にスペースを用いてはならない。アルファベットと数字とハイフンのみ使用可。ソースファイル名の最後は.c とすること。
2. Cソースコード表示モードにする。

1.1.3 エラーの発生の原因 (例)

studio.h 正しくは stdio.h

gcc-lm gcc とハイフンの間にスペースが必要

gcc lm Helo.c ファイル名のタイプミス

2 簡単なプログラムの作成と実行 (教科書第 0 章)

プログラム作成から実行までを、教科書 p. 7 の「リスト 0.1」を例にして述べる。

リスト 0.1 **hello,world** の表示 (ただし, 課題に必要なように変更した)

```
/* プログラミング演習 課題 1 */
/* xxxxxxxx(各自の学生 ID) 各自の氏名 */
#include <stdio.h>

int main(void)
{
    unsigned int i, n;

    printf("input data: ");
    scanf("%d",&n);

    for(i=0;i<n;i++)
    {
        printf("hello, world\n");
    }

    return 0;
}
```

3 C プログラミングの基礎 (教科書第 1 章)

既に教科書にしたがって講義した

4 入出力と演算子 (教科書第 2 章)

4.1 標準入出力 (教科書 2.1 節, 2.2 節)

既に教科書にしたがって講義した

4.2 演算子 (教科書 2.3 節)

既に教科書にしたがって講義した

4.3 数学関数 (教科書 10.2.5 節)

数学関数は、C 言語では、あらかじめ標準的に用意されている。

よく用いられる数学関数を表にしておく

関数	説明
<code>sin(x)</code>	三角関数 $\sin x$ で x の単位はラジアン
<code>cos(x)</code>	三角関数 $\cos x$ で x の単位はラジアン
<code>tan(x)</code>	三角関数 $\tan x$ で x の単位はラジアン
<code>asin(x)</code>	逆三角関数 $\sin^{-1} x$ で x の範囲は $-1 < x < 1$
<code>acos(x)</code>	逆三角関数 $\cos^{-1} x$ で x の範囲は $-1 < x < 1$
<code>atan(x)</code>	逆三角関数 $\tan^{-1} x$
<code>exp(x)</code>	指数関数 e^x
<code>log(x)</code>	自然対数 (底が e の対数) $\ln x$ ($\log_e x$) で x の範囲は $x > 0$
<code>log10(x)</code>	常用対数 (底が 10 の対数) $\log x$ ($\log_{10} x$) で x の範囲は $x > 0$
<code>sqrt(x)</code>	平方根 \sqrt{x} で x の範囲は $x \geq 0$
<code>pow(x,y)</code>	べき乗 x^y
<code>fabs(x)</code>	絶対値 $ x $
<code>ceil(x)</code>	x 以上の最小の整数
<code>floor(x)</code>	x 以下の最大の整数

これらは全て、引数 x, y が `double` 型で、関数値 (戻り値) も全て `double` 型である。 x, y に `double` 型以外の変数を用いた場合、それらを一旦 `double` 型に変換してから関数値が計算される。

先の表にあげた数学関数をプログラム中で使用する場合には、プログラムの冒頭 (`#include <stdio.h>` の直後) に `#include <math.h>` という行を書く必要がある。これによって `math.h` という標準ヘッダファイルが読み込まれる。

このほかにも様々な関数が用意されている。それぞれの関数に対応した標準ヘッダファイルを `include` 文を使って読み込む必要がある。

プログラム例

度 (degree) 単位の角度をキーボードから読み込み, その $\sin x / \cos x$ と $\tan x$ を求め, ディスプレイに表示する。

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

int main(void)
{
    float deg, rad, pai=3.141592;
    float sc, t;

    printf("Input angle in degree: ");
    scanf("%f", &deg);

    rad = deg / 180.0 * pai;
    sc = sin(rad) / cos(rad);
    t = tan(rad);

    printf("Angle in radian is %f\n", rad);
    printf(" sin/cos = %f\n tan    = %f\n", sc, t);

    return 0;
}
```

演習問題

- 4-1. $V = 2.50 \text{ m}^3$ の容器に 1 mol の理想気体が入っている。絶対温度 T を入力すれば圧力 P が Pa 単位で出力されるプログラムを作成せよ。気体定数は $R = 8.31 \text{ J K}^{-1} \text{ mol}^{-1}$ である。 $T = 298.15 \text{ K}$ の場合についてプログラムを実行せよ。
- 4-2. 気体 1 mol に対する van der Waals 状態方程式 $\left(P + \frac{a}{V^2}\right)(V - b) = RT$ を用いて、メタン 1 mol の圧力 P と体積 V を入力して絶対温度 T を求めるプログラムを作成せよ。気体定数は $R = 8.31 \text{ J K}^{-1} \text{ mol}^{-1}$ 、メタンについて $a = 2.283 \times 10^{-1} \text{ m}^6 \text{ Pa mol}^{-2}$ 、 $b = 4.278 \times 10^{-5} \text{ m}^3 \text{ mol}^{-1}$ である。 $P = 1.50 \times 10^6 \text{ Pa}$ 、 $V = 3.0 \times 10^{-3} \text{ m}^3$ の場合についてプログラムを実行せよ。
- 4-3. 一次反応 $A \rightarrow B$ で、反応速度定数を k 、 A の初濃度を c_0 とすると、時刻 t における A の濃度 c は、 $c = c_0 e^{-kt}$ とかける。 $c_0 = 5.00 \text{ mol dm}^{-3}$ 、 $k = 1.53 \times 10^{-1} \text{ s}^{-1}$ の場合について、 t を入力すれば c が出力されるプログラムを作成せよ。 $t = 65.2 \text{ s}$ の場合についてプログラムを実行せよ。
- 4-4. 理想気体を絶対温度 T で等温可逆的に体積 V_1 から V_2 に膨張させたとき、系が外界にした仕事 w は、 $w = RT \ln(V_2/V_1)$ と与えられる。 $R = 8.31 \text{ J K}^{-1} \text{ mol}^{-1}$ は気体定数、 T は絶対温度である。 $T = 298.15 \text{ K}$ の場合について、 V_1 、 V_2 を入力すれば w が出力されるプログラムを作成せよ。 $V_1 = 1.50 \text{ m}^3$ 、 $V_2 = 7.80 \text{ m}^3$ の場合についてプログラムを実行せよ。
- 4-5. 反応速度定数 k に対する Arrhenius の式は、 $k = A \exp(-E_a/RT)$ とかける。 A は前指数因子、 E_a は活性化エネルギー、 $R = 8.31 \text{ J K}^{-1} \text{ mol}^{-1}$ は気体定数、 T は絶対温度である。絶対温度 T_1 のときの速度定数が k_1 、絶対温度 T_2 のときの速度定数が k_2 とする。 T_1 、 k_1 および T_2 、 k_2 を読み込んで、対数プロットの切片と傾きとから A と E_a を求めるプログラムを作成せよ。 $T_1 = 273.15 \text{ K}$ で $k_1 = 3.15 \times 10^{-2} \text{ dm}^3 \text{ mol}^{-1} \text{ s}^{-1}$ 、 $T_2 = 373.15 \text{ K}$ で $k_2 = 4.56 \times 10^{-1} \text{ dm}^3 \text{ mol}^{-1} \text{ s}^{-1}$ の場合についてプログラムを実行せよ。
- 4-6. 整数 m 、 n と実数 a とを読み込み、 $m*a/n$ と $m/n*a$ を計算して画面に出力するプログラムを作れ。色々な数を入力して、何故そのような計算結果になるのか説明せよ。 m 、 n を `cast` を用いて `float` 型に変換してから計算すれば、どのようになるか。
- 4-7. n 本のクジのうち m 本が当たりである。 n 、 m を入力すれば、続けて引いた 2 本のクジが両方当たる確率、片方だけ当たる確率、両方はずれぬ確率を計算せよ。 $n = 561$ 、 $m = 53$ の場合についてプログラムを実行せよ。(ヒント：整数型と実数型に注意が必要)

5 分岐 (教科書第 3.1 節)

教科書にしたがって講義した

演習問題

- 5-1. キーボードから整数 n を読み込み、それが偶数であるか奇数であるかを判断してディスプレイに表示するプログラムを作れ。複数の適当な値について実行してみることに。
- 5-2. 明治元年は 1868 年、大正元年は 1912 年、昭和元年は 1926 年、平成元年は 1989 年である。西暦をキーボードから入力すれば元号に変換してディスプレイに表示するプログラムを作成せよ。ただしその年の 1 月 1 日から新しい元号になったものとする (本当は違う)。また、1867 年以前は範囲外であると表示せよ。1900 年、1925 年、1950 年、自分の生年について実行せよ。
- 5-3. 次の条件 (1), (2) のどちらか片方が満たされるとき、うるう年になる。
(1) 4 の倍数だが 100 の倍数ではない。
(2) 400 の倍数である。

キーボードから西暦の年数 n を読み込み、その年が閏年かどうかを判断するプログラムを作成せよ。思いつく歴史上の重要な年について、実行せよ。例えば平城遷都の年とか、平安遷都の年、ナントの勅令の年など。

- 5-4. ある授業の成績は次の基準で評価されるとする。
- 100 点満点の期末テストの点数を 0.7 倍したものを a とする。
 - 全部で 15 回あったレポートの提出数 n が 10 以上のものは $b = 20$ 、それ以下のものは $b = 2n$ とする。
 - 全部で 15 回あった授業の出席数 m については、 $c = m - 5$ とする。
 - これで $x = a + b + c$ が 100 点満点になる。
 - そこで、 $0 \leq x < 60$ ならば最終成績が F, $60 \leq x < 70$ ならば D, $70 \leq x < 80$ ならば C, $80 \leq x < 90$ ならば B, $90 \leq x \leq 100$ ならば A とする。
 - ただし、 x の値に関わらず、 $m < 10$ の場合には最終成績を F とする。

学籍番号、期末テスト点数、レポート提出数、出席数のデータを、キーボードから順に入力し、ディスプレイに学籍番号、合計点数 x 、最終成績 (A, B, C, D, F のいずれか) を表示するプログラムを作成せよ。複数の適当な例について実行してみることに。

- 5-5. 3 辺の長さが a, b, c である三角形の面積 S は次のように与えられる。

$$S = \sqrt{w(w-a)(w-b)(w-c)}$$

$$w = \frac{a+b+c}{2}$$

キーボードから a, b, c を入力して、三角形の面積を計算し結果を表示するプログラムを作成せよ。ただし、三角形にならない場合には「三角形になりません」というメッセージを表示することにする。複数の適当な例について実行してみることに。

- 5-6. 定形外の国際郵便料金は以下の通りであるとする (実際とは少し違う)。

	第一地域 アジア等	第二地域 北米等	第三地域 欧州等
50 g まで	220 円	260 円	300 円
50 g を越える 50 g ごとに	110 円増し	140 円増し	180 円増し

キーボードから地域番号と重量を入力し、地域番号、重量、料金をディスプレイに表示する操作を繰り返す。

返すプログラムを作成せよ。地域番号に 1, 2, 3 以外の数字を入力した場合にプログラムを終了するものとする。複数の適当な例について実行してみること。

5-7. 方程式 $ax^2 + bx + c = 0$ がある。キーボードから 3 つの係数 a, b, c を入力し、下に示したような答えをディスプレイに表示するプログラムを作成せよ。出力する答えは次のとおりである

- $a = 0$ の場合は一次方程式の解
- $a \neq 0$ の場合で実根を持つときは二次方程式の解
- $a \neq 0$ の場合で虚根を持つときは '虚根です' というメッセージ
- $a = 0$ かつ $b = 0$ の場合は '解なし' というメッセージ

次の 5 組の係数についてプログラムを実行せよ。

- $(a = 3, b = 4, c = -4)$
- $(a = 5, b = -2, c = 10)$
- $(a = 1, b = 6, c = -9)$
- $(a = 6, b = -15, c = -50)$
- $(a = 1, b = 1, c = -6)$

6 繰り返し (教科書 3.2 節)

教科書にしたがって講義した

演習問題

- 6-1. 次の数列の和 r, s, t を計算するプログラムを作成し, $n = 7$ の場合について実行せよ。 n の値はキーボードから入力し, 計算結果を画面に表示せよ。

$$r = \sum_{j=1}^n \frac{j}{j+1}, \quad s = \sum_{j=1}^n \frac{2j-1}{3j}, \quad t = \sum_{j=1}^n \frac{2j}{j^2+1}$$

- 6-2. 次の数列の和 u, v, w を計算するプログラムを作成し, $n = 7$ の場合について実行せよ。 n の値はキーボードから入力し, 計算結果を画面に表示せよ。

$$u = \sum_{j=1}^n (-1)^{j+1} j, \quad v = \sum_{j=1}^n (-1)^{j+1} \left(\frac{2j-1}{j^3+1} \right), \quad w = \sum_{j=1}^n \left(-\frac{1}{2} \right)^{j-1}$$

- 6-3. 次の Clausius-Clapeyron の関係式を用いて 50°C から 100°C まで 5°C おきに水の蒸気圧を求め, 各温度とその温度に対応する蒸気圧を画面に出力せよ。 ($R = 8.31 \text{ J K}^{-1} \text{ mol}^{-1}$, $\Delta H = 40.65 \text{ kJ mol}^{-1}$, $T_0 = 373.15 \text{ K}$, $P_0 = 1.01325 \times 10^5 \text{ Pa}$)

$$\ln \left(\frac{P}{P_0} \right) = -\frac{\Delta H}{R} \left(\frac{1}{T} - \frac{1}{T_0} \right)$$

- 6-4. 金利年 3% (固定金利) で 2000 万円を借り入れた。年間 100 万円ずつ返却すると何年間ですべて返却できるか計算しなさい。各年度毎の残高表も出力すること。ただし, 1 円未満は四捨五入し, 金利は借り入れた時点で付くもの (例えば, 1 年目の残高 $2000 \times 1.03 - 100 = 1960$ 万円) として計算しなさい。例えば返却期間を多く取って残高がマイナスになった時点ですべて返却できたものとすればよい。
- 6-5. 0.01 mol dm^{-3} 塩酸 25 cm^3 を 0.01 mol dm^{-3} 水酸化ナトリウム溶液で滴定する。加える水酸化ナトリウムの量を 0 cm^3 から 50 cm^3 まで 1 cm^3 ずつ変化させたときのそれぞれの pH の値を求めなさい。ただし, 水のイオン積 K_w は $1.01 \times 10^{-14} \text{ mol}^2 \text{ dm}^{-6}$ とする。結果は表にして画面に出力せよ。
- 6-6. 酢酸の酸解離定数 K_a は $1.7 \times 10^{-5} \text{ mol dm}^{-3}$ である。酢酸の濃度を mol dm^{-3} の単位で 1.00×10^{-1} , 1.00×10^{-2} , 1.00×10^{-3} , \dots , 1.00×10^{-9} , 1.00×10^{-10} と変えたときのそれぞれの電離度 (または解離度) α を求めなさい。結果は表にして画面に出力せよ。
- 6-7. 異なる二つの正の整数 m と n をキーボードから読み込み, 次の方法でそれらの最大公約数を求めて画面に表示するプログラムを作成せよ。計算の途中経過も工夫して表示すること。計算法 (ユークリットの互除法)
- $m > n$ ならば $m = m - n$, $m < n$ ならば $n = n - m$ とする。この計算を繰り返して $m = n$ になったとき, それがはじめの m と n の最大公約数である。
- 6-8. 異なる二つの正の整数 m と n をキーボードから読み込み, それらの最小公倍数 p を求めて画面に表示するプログラムを作成せよ。ただし, p は次の式で計算できる。

$$p = \frac{mn}{f}$$

ここで f は m, n の最大公約数である。 f は 6-7 を参考にして求めること。

- 6-9. x の値を $0.0 \sim 1.0$ まで 0.1 ごとに変化させて, 次の式にしたがって e^x の値を計算するプログラムを作成し実行せよ。

$$e^x = 1 + x + \frac{x^2}{2!} + \dots = \sum_{j=0}^{\infty} \frac{x^j}{j!}$$

もしも

$$e^x = \sum_{j=0}^{\infty} a_j$$

と書けば、各項の関係が次のように書けることを利用すること。

$$a_j = \frac{x}{j} a_{j-1}, \quad j \geq 1$$

a_j の絶対値が 1.0×10^{-6} より小さくなった場合に和の計算を打ち切り、それまでの項の和を計算結果とする。結果は見やすい数表の形で画面に表示させること。二重ループのプログラムになるのでじっくり考えること。

6-10. かけ算の九九の表を作りなさい。二重ループで作成すること。書式を工夫すること。

7 関数 (教科書第 4 章)

教科書にしたがって講義した

演習問題

- 7-1. 演習問題 6-1 (p. 13) で, n を引数として数列の和 r, s, t を計算する関数, `rcalc(n)`, `scalc(n)`, `tcalc(n)` を作り, `main` 関数からその関数を呼び出すようなプログラムに作り直しなさい。自作関数の戻り値および関数内の計算は `float` 型とする。
- 7-2. 演習問題 6-2 (p. 13) で, n を引数として数列の和 u, v, w を計算する関数, `ucalc(n)`, `vcalc(n)`, `wcalc(n)` を作り, `main` 関数からその関数を呼び出すようなプログラムに作り直しなさい。自作関数の戻り値および関数内の計算は `float` 型とする。
- 7-3. 演習問題 6-9 (p. 13) を参考にして, e^x を Taylor 展開に基づいて計算する関数 `myexp(x)` を自作し, `main` 関数内で標準関数である `exp(x)` と `myexp(x)` をそれぞれ呼び出して結果を比較するプログラムに作りなおしなさい。 x の値は 0.0 ~ 1.0 まで 0.1 ごとに変化させること。自作関数の戻り値および関数内の計算は `float` 型とする。
- 7-4. $\sin x$ は次のように Taylor 展開できる。

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} + \dots$$

もしも

$$\sin x = \sum_{j=1}^{\infty} a_j$$

と書き, $a_1 = x$ とすれば, 各項の関係が次のように書ける。

$$a_{j+1} = -\frac{x^2}{2j(2j+1)}a_j, \quad j \geq 1$$

これを利用して, $\sin x$ の値を計算する関数 `mysin(x)` を自作せよ。自作関数の戻り値および関数内の計算は `float` 型とする。さらに, 標準関数である `sin(x)` と `mysin(x)` をそれぞれ `main` 関数内で呼び出して結果を比較するプログラムを作りなさい。 x の値は 0.0 ~ 1.0 まで 0.1 ごとに変化させること。結果は見やすい数表の形で画面に表示させること。

- 7-5. $\cos x$ は次のように Taylor 展開できる。

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} + \dots$$

もしも

$$\cos x = \sum_{j=1}^{\infty} a_j$$

と書き, $a_1 = 1$ とすれば, 各項の関係が次のように書ける。

$$a_{j+1} = -\frac{x^2}{2j(2j-1)}a_j, \quad j \geq 1$$

これを利用して, $\cos x$ の値を計算する関数 `mycos(x)` を自作せよ。自作関数の戻り値および関数内の計算は `float` 型とする。さらに, 標準関数である `cos(x)` と `mycos(x)` をそれぞれ `main` 関数内で呼び出して結果を比較するプログラムを作りなさい。 x の値は 0.0 ~ 1.0 まで 0.1 ごとに変化させること。結果は見やすい数表の形で画面に表示させること。

8 配 列 (教科書第 6 章)

教科書にしたがって講義した

演習問題

- 8-1. 正の整数 n をキーボードから入力し、次に述べる「エラトステネスのふるい」と呼ばれる方法を用いて、 $2 \sim n$ までの整数のうち素数であるものをすべて求め、画面に表示しなさい。

計算法

まず、十分大きな整数型の配列（例えば `int furui[1001]`；とする、当然 $n \leq 1000$ でなければならぬ）を用意し、`furui[i]` に整数 i を入力しておく。 $m = 2$ として、`furui[i]` が m で割り切れれば `furui[i]` をゼロにする。 $m = 3 \sim \sqrt{n}$ について、`furui[i]` がゼロではないとき、 m で割り切れれば `furui[i]` をゼロにする。これらの操作が終わった後で `furui[i]` のうちゼロでないものが素数である。

- 8-2. 後で述べる方法を用いて整数の疑似乱数を 100 個発生させ、それを整数型の配列（例えば `int iransu[101]`；とする）に入れる。その後、100 個の数の平均と分散を実数型で求めよ。分散は二乗の平均から平均の二乗を引いたものである。

疑似乱数の発生（線形合同法）

例えば $A=12367, B=57, M=467$ とし、`iransu[1]=7` とする。次の漸化式を用いれば、疑似乱数が発生できる。`iransu[i+1]=(A * iransu[i] + B) % M` ただし、`%` は余りを求める算術演算子である。

- 8-3. n 行 m 列の行列 \mathbf{A} を 2 次元配列に代入し、 \mathbf{A} とその転置行列 ${}^t\mathbf{A}$ を画面に出力するプログラムを作成せよ。
- 8-4. 1 行目にテストの受験者数、2 行目以降にはテストの点数が各行に 1 つずつ書かれたデータファイルがある。このファイルから、点数を配列として読み込み、最高点、最低点、平均点、標準偏差をもとめて画面に表示するプログラムを作成しなさい。ただし、受験者数は最大 300 名であるとする。標準偏差 σ は、2 乗の平均から平均の 2 乗を引いたものの平方根で、次のように与えられる。

$$\sigma = \sqrt{\langle x^2 \rangle - \langle x \rangle^2}, \quad \langle x^2 \rangle = \frac{1}{n} \sum_{i=1}^n x_i^2, \quad \langle x \rangle = \frac{1}{n} \sum_{i=1}^n x_i$$

- 8-5. 前問 8-4 のプログラムを変更し、各人の偏差値を計算して画面に表示する操作を付け加えよ。各人の偏差値 h_i は次のように与えられる。

$$h_i = 50.0 + \frac{10.0 \cdot (x_i - \langle x \rangle)}{\sigma}$$

- 8-6. 1 行目にデータの組の数 n 、2 行目以降の各行に x_i と y_i がコンマで区切られて書かれたデータファイルがある。このファイルから、 n, x_i, y_i を読み込み、 x_i と y_i ($1 \leq i \leq n$) をそれぞれ配列に記憶する。このデータをもとに、最小二乗法によって x_i と y_i を直線にあてはめた場合の、切片 a と傾き b および、偏差の二乗和 S を画面に表示するプログラムを作成して実行せよ。

$$a = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{n \sum x_i^2 - (\sum x_i)^2}$$

$$b = \frac{n \sum y_i x_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2}$$

$$S = \sum_{i=1}^n (a + b x_i - y_i)^2$$

8-7. 次の行列計算を求めるプログラムを作成せよ。ベクトル \mathbf{x} , \mathbf{y} , はそれぞれ (n) , (m) の 1 次元配列, 行列 \mathbf{A} は (m, n) の 2 次元配列の変数である。結果はディスプレイに表示すること。

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_m \end{pmatrix}$$

ただし,

$$y_i = \sum_j A_{ij}x_j$$

8-8. 次の行列計算を求めるプログラムを作成せよ。行列 \mathbf{A} , \mathbf{B} , \mathbf{C} はそれぞれ (n, l) , (l, m) , (n, m) の 2 次元配列の変数である。結果はディスプレイに表示すること。

$$\begin{pmatrix} A_{11} & \cdots & A_{1l} \\ A_{21} & \cdots & A_{2l} \\ \vdots & \cdots & \vdots \\ A_{n1} & \cdots & A_{nl} \end{pmatrix} \begin{pmatrix} B_{11} & \cdots & B_{1m} \\ B_{21} & \cdots & B_{2m} \\ \vdots & \cdots & \vdots \\ B_{l1} & \cdots & B_{lm} \end{pmatrix} = \begin{pmatrix} C_{11} & \cdots & C_{1m} \\ C_{21} & \cdots & C_{2m} \\ \vdots & \cdots & \vdots \\ C_{n1} & \cdots & C_{nm} \end{pmatrix}$$

ただし,

$$C_{ij} = \sum_k A_{ik}B_{kj}$$

8-9. 次のようなデータ A_{ij} ($i = 1, 2, \dots, m, j = 1, 2, \dots, n$) がある。

```

39  85  88  32  81  20
14  16  49  40  10  54
24  68  26  99  85  81
24  73   3  48  28  24
67  18  77  57  68  23
56  44  62  43  56  54
16  51  69  38  79  89
37  59  94  11  75  26

```

これを読み込み, その行和 G_i , 行平均 g_i , 列和 R_j , 列平均 r_j , 総和 T , 総平均 t を計算せよ。

$$G_i = \sum_{j=1}^n A_{ij}, \quad g_i = G_i/n$$

$$R_j = \sum_{i=1}^m A_{ij}, \quad r_j = R_j/n$$

$$T = \sum_{i=1}^m \sum_{j=1}^n A_{ij}, \quad t = T/(mn)$$

出力は下記のように行え。

$i \setminus j$	1	2	...	n	行和	行平均
1	A_{11}	A_{12}	...	A_{1n}	G_1	g_1
2	A_{21}	A_{22}	...	A_{2n}	G_2	g_2
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\vdots
m	A_{m1}	A_{m2}	...	A_{mn}	G_m	g_m
列和	R_1	R_2	...	R_n	T	
列平均	r_1	r_2	...	r_n		t

9 数値積分

x_0 から x_n まで、 h 間隔に $(n+1)$ 個のデータ点があり、 $f(x_0), f(x_1), \dots, f(x_n)$ がわかっているとして、次の定積分を計算したい。

$$I = \int_{x_0}^{x_n} f(x) dx \quad (1)$$

9.1 台形法

$f(x_i)$ と $f(x_{i+1})$ を直線で結んで近似する。個々の区間は台形になるので、面積が簡単に計算できる。

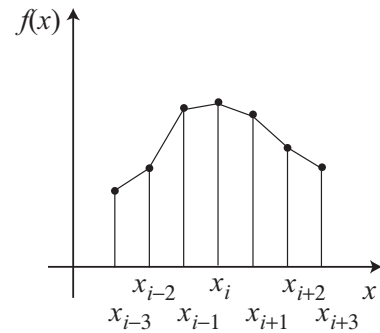
$$J_i = \frac{h}{2} (f(x_i) + f(x_{i+1})) \quad (2)$$

すると、全範囲の積分は次のように書ける。

$$I = \int_{x_0}^{x_n} f(x) dx = \sum_{i=0}^{n-1} J_i \quad (3)$$

したがって、全体の積分は次のように計算できる。

$$I = \frac{h}{2} \left(f(x_0) + 2 \sum_{i=1}^{n-1} f(x_i) + f(x_n) \right) \quad (4)$$



9.2 シンプソン法

$f(x_{i-1}), f(x_i), f(x_{i+1})$ の 3 点を 2 次関数で結んで近似する。 n が偶数である必要がある。

$$f(x) = ax^2 + bx + c \quad (5)$$

$$\begin{aligned} K_i &= \int_{x_{i-1}}^{x_{i+1}} f(x) dx \\ &= \int_{x_{i-1}}^{x_{i+1}} (ax^2 + bx + c) dx \\ &= \left[\frac{a}{3} x^3 + \frac{b}{2} x^2 + cx \right]_{x_{i-1}}^{x_{i+1}} \end{aligned} \quad (6)$$

ただし、 $x_{i-1} = x_i - h$ であり、また $x_{i+1} = x_i + h$ なので

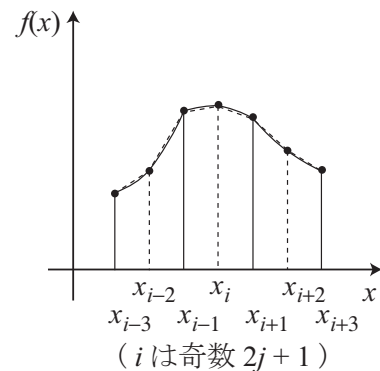
$$\begin{aligned} K_i &= \frac{a}{3} ((x_i + h)^3 - (x_i - h)^3) + \frac{b}{2} ((x_i + h)^2 - (x_i - h)^2) \\ &\quad + c((x_i + h) - (x_i - h)) \\ &= \frac{2a}{3} (hx_i^2 + h^3) + bhx_i + 2ch \\ &= \frac{h}{3} (6ax_i^2 + 2ah^2 + 6bx_i + 6c) \end{aligned} \quad (7)$$

また

$$f(x_{i-1}) = a(x_i - h)^2 + b(x_i - h) + c \quad (8)$$

$$f(x_i) = ax_i^2 + bx_i + c \quad (9)$$

$$f(x_{i+1}) = a(x_i + h)^2 + b(x_i + h) + c \quad (10)$$



そこで、次のような計算を行う

$$f(x_{i-1}) + 4f(x_i) + f(x_{i+1}) = 6ax_i^2 + 2ah^2 + 6bx_i + 6c \quad (11)$$

これは (7) 式の 3 行目のカッコ内と同じなので、 K_i が次のように書けることになる。

$$K_i = \frac{h}{3} (f(x_{i-1}) + 4f(x_i) + f(x_{i+1})) \quad (12)$$

すると、全範囲の積分は次のように書ける。

$$I = \int_{x_0}^{x_n} f(x) dx = \sum_{j=0}^{(n/2)-1} K_{2j+1} \quad (13)$$

つまり、奇数番目を中心にした K_i をすべてたし合わせることになる。(12) 式と (13) 式をまとめる。

$$I = \frac{h}{3} \left(f(x_0) + 4 \sum_{k=1}^{n/2} f(x_{2k-1}) + 2 \sum_{k=1}^{(n/2)-1} f(x_{2k}) + f(x_n) \right) \quad (14)$$

カッコの中は、はじめと終わりの点はそのまま、奇数番目は 4 倍、偶数番目は 2 倍して和をとることになる。これをシンプソンの公式という。

演習問題

9-1. 次の積分を $n = 200$ とした台形法で計算するプログラムを作成せよ。

$$\int_0^{\pi} \sin x \, dx$$

9-2. 次の積分を $n = 200$ としたシンプソン法で計算するプログラムを作成せよ。

$$\int_0^{\pi} \cos x \, dx$$

9-3. π は次のような積分で求めることができる。

$$\int_0^2 \sqrt{4-x^2} \, dx = \pi$$

$n = 100$ とした台形法でこの積分を数値計算し、 π を求めるプログラムを作成せよ。

9-4. 前問 9-3 で n をかえて計算した場合、 π が 6 桁目まで正確に得られるのは n がいくらの場合か確かめよ。

9-5. π は次のような積分で求めることができる。

$$\int_0^1 \frac{4}{1+x^2} \, dx = \pi$$

$n = 100$ としたシンプソン法でこの積分を数値計算し、 π を求めるプログラムを作成せよ。

9-6. 前問 9-5 で n をかえて計算した場合、 π が 6 桁目まで正確に得られるのは n がいくらの場合か確かめよ。

9-7. 次の積分を 6 桁の精度まで台形法で数値計算するプログラムを作成せよ。

$$\int_0^{\infty} e^{-x} \, dx$$

9-8. 次の積分を 6 桁の精度までシンプソン法で数値計算するプログラムを作成せよ。

$$\int_0^{\infty} e^{-x^2} \, dx$$

10 非線形方程式の数値解法

実在系の状態方程式から p - V - T の関係を知りたい場合や、波動関数の節の位置を知りたいときなどに非線形方程式を解く必要が生じる。ここでは、そのような場合に用いる数値的方法の原理を学ぶ。

10.1 二分法の原理

次の方程式の解を求めたい。

$$f(x) = 0 \quad (15)$$

1. $f(a_0)$ と $f(b_0)$ が異なる符号をもつように、初期値 a_0 と b_0 を決める。
式で書くと次の条件になる。

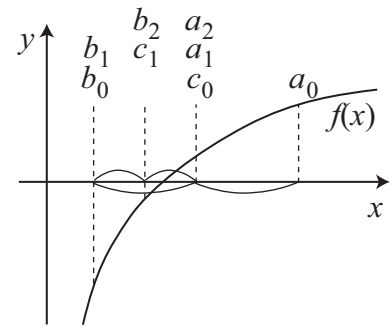
$$f(a_0)f(b_0) < 0 \quad (16)$$

2. a_0 と b_0 の中点を c_0 とする。

$$c_0 = \frac{a_0 + b_0}{2} \quad (17)$$

3. はじめに $i = 1$ とする。
4. 次のように場合分けする。

- $f(a_{i-1})$ と $f(c_{i-1})$ が異符号 ($f(a_{i-1})f(c_{i-1}) < 0$) なら、 a_{i-1} と c_{i-1} の間に解が存在する。この場合、 $a_i = a_{i-1}$ 、 $b_i = c_{i-1}$ とする。
- $f(a_{i-1})$ と $f(c_{i-1})$ が同符号 ($f(a_{i-1})f(c_{i-1}) > 0$) なら、 $f(c_{i-1})$ と $f(b_{i-1})$ が異符号 ($f(c_{i-1})f(b_{i-1}) < 0$) のはずであり、その間に解が存在する。この場合、 $a_i = c_{i-1}$ 、 $b_i = b_{i-1}$ とする。



5. a_i と b_i の中点を c_i とする。

$$c_i = \frac{a_i + b_i}{2} \quad (18)$$

6. 次の 2 つの条件がどちらも満たされていないければ、 i を 1 増やし、(4) にもどって計算を繰り返す。

$$|f(c_i)| < \delta \quad (19)$$

$$|a_i - b_i| < \varepsilon \quad (20)$$

δ と ε は求める計算精度によって決まる正の定数である。

10.2 ニュートン法の原理

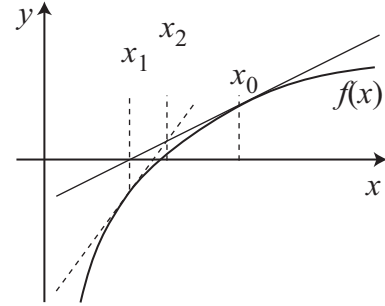
ニュートン法は、 $f(x)$ の導関数 $f'(x)$ がわかっているときに用いることのできる、効率的な方法である。

1. はじめに初期値 x_0 を適当に選ぶ。
2. $i = 1$ とする。
3. $x = x_{i-1}$ における $f(x)$ の接線と x 軸との交点を x_i とする。接線の方程式は次のように与えられる。

$$y = f'(x_{i-1})(x - x_{i-1}) + f(x_{i-1}) \quad (21)$$

したがって、 x_i は次のように与えられる。

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})} \quad (22)$$



4. 次の条件が満たされていないならば、 i を 1 増やし、(3) にもどって計算を繰り返す。

$$|f(x_i)| = \delta \quad (23)$$

演習問題

- 10-1. 二分法によって次の方程式を解くプログラムを作成し、初期値が $a_0 = 0$, $b_0 = 3$ の場合について実行せよ。ただし $\delta = 10^{-6}$, $\varepsilon = 10^{-6}$ とすること。途中経過と最終結果がわかるように画面に表示すること。

$$f(x) = 5x^3 - 7x^2 + x - 9 = 0$$

- 10-2. Newton 法によって次の方程式を解くプログラムを作成し、初期値が $x_0 = 3$ の場合について実行せよ。ただし $\delta = 10^{-6}$ とすること。途中経過と最終結果がわかるように画面に表示すること。

$$f(x) = 5x^3 - 7x^2 + x - 9 = 0$$

- 10-3. 二分法によって次の方程式を解くプログラムを作成し、初期値が $a_0 = 6$, $b_0 = 9$ の場合について実行せよ。ただし $\delta = 10^{-6}$, $\varepsilon = 10^{-6}$ とすること。途中経過と最終結果がわかるように画面に表示すること。

$$f(x) = \frac{x \sin x}{2} + x - 9 = 0$$

- 10-4. Newton 法によって次の方程式を解くプログラムを作成し、初期値が $x_0 = 6$ の場合について実行せよ。ただし $\delta = 10^{-6}$ とすること。途中経過と最終結果がわかるように画面に表示すること。

$$f(x) = \frac{x \sin x}{2} + x - 9 = 0$$

- 10-5. 二分法によって次の方程式を解くプログラムを作成し、適当な初期値を用いてプログラムを何度か走らせ、すべての解を求めなさい。解は $5 < x < 15$ の範囲に 3 つある。ただし $\delta = 10^{-6}$, $\varepsilon = 10^{-6}$ とすること。

$$f(x) = \frac{x \sin x}{2} + x - 9 = 0$$

- 10-6. Newton 法によって次の方程式を解くプログラムを作成し、適当な初期値を用いてプログラムを何度か走らせ、すべての解を求めなさい。解は $5 < x < 15$ の範囲に 3 つある。ただし $\delta = 10^{-6}$ とすること。

$$f(x) = \frac{x \sin x}{2} + x - 9 = 0$$

11 ファイル処理 (教科書第 9 章)

これまで、数値の入力はキーボードから、計算結果の出力はディスプレイに行ってきた。しかし、これでは大量のデータを入力する場合に効率が悪く確実性にも欠ける。また、結果を保存する手だても必要である。そのため、入出力にディスク上のファイルを使用することが多い。ここで説明するのは、必要最小限の最も簡単な形のファイル操作である。内容を詳しく説明すると難しくなるが、まずは「オマジナイ」としてやり方を覚えて欲しい。よりきめ細かい操作に関しては、教科書第 9 章 (p. 207) を参照すること。

11.1 計算結果をファイルに保存

```
#include <stdio.h>
int main(void)
{
    float a, b, c;
    FILE *fp;          /* ファイルポインタ変数の宣言「オマジナイ」 */

    printf("Input a: "); scanf("%f", &a);
    printf("Input b: "); scanf("%f", &b);
    c = a + b;
    printf("a + b = %f\n",c);

    fp = fopen("abc.dat","w"); /* ファイルオープン「オマジナイ」 */

    fprintf(fp,"a = %f\n",a); /* ファイルに書き込み */
    fprintf(fp,"b = %f\n",b);
    fprintf(fp,"a + b = %f\n",c);

    fclose(fp);          /* ファイルクローズ「オマジナイ」 */

    return 0;
}
```

実行結果

[画面上]

```
input a:  1.2
input b:  2.3
a + b = 3.500000
```

[ファイル]

```
a = 1.200000
b = 2.300000
a + b = 3.500000
```

作成されたファイルを見るには

- BCPad でメニュー「File」 → 「ファイルを開く」
- 「ファイルを開く」ダイアログボックスで「場所」確認（プログラムを保存した場所 programming）
- 「ファイルの種類」の「全てのファイル (*.*)」を選択
- 表示されたファイルから目的の「abc.dat」を選択 → 「OK」

作成したファイルのウィンドウを開いたまま、再びプログラムを実行すると

「ファイル... は他のアプリケーションによって書き換えられました。読み直しますか？」 → 「はい」

解説

```
FILE *fp;
```

ファイルポインタ変数 `fp` を使用することを宣言する。`fp` の前のアスタリスク `*` は `fp` がポインタ変数であることを表す。`FILE` はファイル構造体を表す。`FILE` は `stdio.h` で定義されているので、ファイルを使用する場合には必ず、プログラムの冒頭に `#include <stdio.h>` がなければならない。教科書 p. 206 に詳しい解説がある。

以上の説明は、現時点では何のことか皆目わからないと思う。それで当然である。ここでは、とにかくオマジナイとしてこの行を宣言文の中に書く。

```
fp = fopen("abc.dat", "w")
```

ファイルを使うために、まずファイルを開く。書き込み、読み込み前に必ずオープンしなければならない。これもオマジナイだが、`"abc.dat"` はファイル名なので、許される範囲で適当につける。`"w"` の部分は「アクセス形式」を表す。当面、ファイルに書き込みたい時には `"w"` と書くと思って欲しい。

この書き方ならプログラムを保存している `H:\programming` のフォルダにファイルが作られる。もしも `A:` ドライブのフロッピーに書き込みしたければ次のようにする。

```
fp = fopen("A:\abc.dat", "w")
```

うまく開ければ変数 `fp` にファイルのポインタが代入される。うまくいかなかったときの処理はここではやらない。

```
fprintf(fp, "a = %f\n", a);
```

ここで、実際にファイルへの書き込みが行われる。`printf` によく似た関数 `fprintf` を使う。`printf` との違いは、書式指定（ここでは `"a = %f\n"`）の前に、書き込みを行うファイルのファイルポインタ `fp` を指定しなければならないことである。書式指定の方法は `printf` と全く同じ。

```
fclose(fp);
```

ファイルを閉じる。書き込みが全て終わればファイルを閉じる。書き込みのたびに開いたり閉じたりしたら、そのたびにそれまであった内容が破棄される（アクセス形式 `"w"` の場合）ので注意すること。

11.2 ファイルを用いてデータ入力

```
#include <stdio.h>
int main(void)
{
    float a, b, c;
    FILE *fp1, *fp2;

    fp1 = fopen("input.dat", "r");

    fscanf(fp1, "%f", &a);
```

```

fscanf(fp1,"%f", &b);

fclose(fp1);

c = a + b;

printf("a = %10.5f\nb = %10.5f\n",a,b);
printf("a + b = %10.5f\n",c);

fp2 = fopen("results.dat","w");

fprintf(fp2,"a = %10.5f\nb = %10.5f\n",a,b);
fprintf(fp2,"a + b = %10.5f\n",c);

fclose(fp2);

return 0;
}

```

入力用データファイルの作成

- BCPad で「メニュー」→「File」→「新規作成」
- 新規ファイルのウィンドウが開くので次のようにデータを書き込む
1.2
2.3
- 「メニュー」→「File」→「名前を付けて保存」
- 「名前をつけて保存」ダイアログボックスで「ファイル名を指定」（この場合 `input.dat` とする）→「保存」
- この例のように、ファイル名で拡張子（`.dat` の部分）まで指定すると、「ファイルの種類」の記述に関係なく、ファイル名で指定した拡張子のファイルが作られる。
- ファイル名のみ指定した場合、プログラムファイルと同じフォルダに保存される
- 別のフォルダに保存した場合は「保存する場所」を変更すること

実行結果

画面に次の結果がも表示され、ファイル `results.dat` にも同様に書き込まれる

```

a =    1.20000
b =    2.30000
a + b =   3.50000

```

解説

```
FILE *fp1, *fp2;
```

入力用、出力用の 2 つのファイルを使うので、ファイルポインタ変数も 2 つ宣言した。

```
fp1 = fopen("input.dat","r");
```

読み込み用にファイルをオープンする。書き込み用と異なるところは、アクセス形式が `"r"` になっているところ。

プログラムを保存している H:\programming 以外のフォルダにファイルがある場合の書き方は、書き込み用オープンの場合と同じ。ファイルがうまく開ければ変数 `fp1` にファイルのポインタが代入される。ファイルが存在しなければ `fp1` には `NULL` が代入される。

```
fscanf(fp1,"%f", &a);  
fscanf(fp1,"%f", &b);
```

ここで、実際にファイルからの読み込みが行われる。`scanf` によく似た関数 `fscanf` を使う。`scanf` との違いは、書式指定（ここでは "%f"）の前に、書き込みを行うファイルのファイルポインタ `fp1` を指定しなければならないことである。書式指定の方法は `scanf` と全く同じ。

当然の事ながら、ここで指定した書式とデータファイル `input.dat` の内容が対応していなければならない。

```
fclose(fp1);
```

ファイルのクローズ。`fopen` でオープンしてから `fclose` でクローズするまでの間は、次々とファイル中のデータを読み込んでいく。一旦クローズして再びオープンすると、また先頭に戻ってしまう。

12 ソート

データを小さい順や大きい順に並べ替えることを「ソート」という。ソートには、単純なアルゴリズムで演算速度気にしない方法でも、大きく分けて「選択法」、「バブル法」、「挿入法」の3つの方法がある。以下に、 n 個のデータを小さい順にソートする場合について、各方法を説明する。大きい順にソートする場合に変更するのは簡単なので、各自で考えること。

12.1 選択法

n 個のデータが一次元配列 $a[0] \sim a[n-1]$ に代入されている。このうち $a[0] \sim a[j-1]$ は、すでに n 個のデータの中で小さいものから $j-1$ 番目までのものが順に並んでいるとする。そこで、 $a[j] \sim a[n-1]$ のうちで最小のものを $a[j]$ と入れかえれば j 番目までのソートが完了したことになる。このような入れかえの操作を $j=0 \sim n-2$ の範囲で繰り返せば、 n 個のデータのソートが完了する。

具体的には、 $j=0 \sim n-2$ の範囲で j を1ずつ変化させて次の操作を行う。

1. $mini=a[j]; mnum=j$ とする。
2. $i=j+1 \sim n-1$ の範囲で i を1ずつ変化させて、次の操作を行う。
 - $a[i] < mini$ の場合
 $mini=a[i]; mnum=i$ とする。
 - $a[i] \geq mini$ の場合
何もしない。
 - いずれの場合も次の i へ進む。
3. $a[j]$ と $a[mnum]$ を入れかえる。これで $a[0] \sim a[j]$ のソートが完了する。
4. 次の j に進む。

12.2 バブル法

n 個のデータが一次元配列 $a[0] \sim a[n-1]$ に代入されている。隣り合う2つのデータを比較して小さい方が前にくるように置きかえる。この操作を繰り返せば後ろから順に並べかえが進む。 n 個のデータのソートが完了するまで、これを繰り返す。

具体的には、 $j=0 \sim n-2$ の範囲で j を1ずつ変化させて次の操作を行う。

1. $i=0 \sim n-2-j$ の範囲で i を1ずつ変化させて、次の操作を行う。
 - $a[i] > a[i+1]$ の場合
 $a[i]$ と $a[i+1]$ を入れかえる。
 - $a[i] \leq a[i+1]$ の場合
何もしない。
 - いずれの場合も次の i へ進む。
2. $i=n-2-j$ までいけば、 $a[n-2-j] \sim a[n-1]$ の範囲で並べかえが終了する。
3. 次の j に進む。

12.3 挿入法

n 個のデータが $a[0] \sim a[n-1]$ に代入されている。このうち $a[0] \sim a[j-1]$ は、 j 個のデータ内では小さい順

に並んでいるとする。この並びの中の適当な位置に $a[j]$ を挿入する。このような挿入の操作を $j=1\sim n-1$ の範囲で繰り返せば、 n 個のデータのソートが完了する。

具体的には、 $j=1\sim n-1$ の範囲で j を 1 ずつ変化させて次の操作を行う。

1. $dummy=a[j]$ とする。
2. $i=[j-1]\sim 0$ の範囲で i を -1 ずつ変化させて、次の操作を行う。
 - $a[i] \geq dummy$ の場合
 $a[i+1]=a[i]$ とする。
一旦 $a[i]=dummy$ とする。
次の i へ進む。
 - $a[i] < dummy$ の場合
これで $a[0]\sim a[j]$ の範囲のソートが完了したことになるので、 i のループの外に出る。
3. 次の j に進む。

演習問題

- 12-1. ファイルから n 個のデータを読み込み、選択法を用いて小さい順にソートするプログラムを作成せよ。まず、ソート前のデータを見やすい表にして表示し、その後、1行あけてソート後のデータを同様の表を表示すること。また、画面に表示した内容は、`sort-1.dat` にも保存すること。
- 12-2. ファイルから n 個のデータを読み込み、バブル法を用いて小さい順にソートするプログラムを作成せよ。まず、ソート前のデータを見やすい表にして表示し、その後、1行あけてソート後のデータを同様の表を表示すること。また、画面に表示した内容は、`sort-2.dat` にも保存すること。
- 12-3. ファイルから n 個のデータを読み込み、挿入法を用いて小さい順にソートするプログラムを作成せよ。まず、ソート前のデータを見やすい表にして表示し、その後、1行あけてソート後のデータを同様の表を表示すること。また、画面に表示した内容は、`sort-3.dat` にも保存すること。
- 12-4. ファイルから n 個のデータを読み込み、選択法を用いて大きい順にソートするプログラムを作成せよ。まず、ソート前のデータを見やすい表にして表示し、その後、1行あけてソート後のデータを同様の表を表示すること。また、画面に表示した内容は、`sort-4.dat` にも保存すること。
- 12-5. ファイルから n 個のデータを読み込み、バブル法を用いて大きい順にソートするプログラムを作成せよ。まず、ソート前のデータを見やすい表にして表示し、その後、1行あけてソート後のデータを同様の表を表示すること。また、画面に表示した内容は、`sort-5.dat` にも保存すること。
- 12-6. ファイルから n 個のデータを読み込み、挿入法を用いて大きい順にソートするプログラムを作成せよ。まず、ソート前のデータを見やすい表にして表示し、その後、1行あけてソート後のデータを同様の表を表示すること。また、画面に表示した内容は、`sort-6.dat` にも保存すること。
- 12-7. 二次元配列 $a[2][n]$ に n 人分の成績データがある。 $a[0][0]\sim a[0][n-1]$ には学籍番号、 $a[1][0]\sim a[1][n-1]$ にはテストの点数が記憶されている。このデータを、学籍番号とテストの点数の関係を保ったままで、テストの成績順にソートするプログラムを作成せよ。