

配列 (応用 2)

山本昌志*

2007 年 1 月 12 日

概要

前回は引き続きで配列を使ったプログラミングを学習する。ファイルから配列にデータを格納する方法と、関数の引数での配列の取扱いについて学ぶ。

1 前回の復習と本日の内容

1.1 前回の復習

1.1.1 プログラム例

前回の講義では、ファイルからデータを読み込んで配列に格納し、配列の内容を関数に受渡して、処理することを学んだ。ここで、理解すべき内容は、

- ファイルからデータを読み込んでファイルに格納する方法
- ユーザー定義関数に配列データの受渡す方法。

のふたつである。

それぞれについて、リスト 1 のプログラムを例にして説明する。このプログラムは、つぎに示すデータをファイルから読み込んで、全ての整数の合計値と各列の平均値を計算するプログラムである。データは、10000 行 4 列、合計 4 万個の整数である。

```
-76797 99987 53528 -43172
-34698 44783 -106207 -106631
-83424 31615 18774 -4134
78694 -886 64632 103022
-86516 -99744 -51044 -11396
90058 54995 -39364 -36610
-78969 106494 -5209 -95276
75913 32002 39260 106490
24615 -14585 -44056 97291
-77175 -42889 98034 -53226
96100 9434 50013 67420
```

この辺は長いので省略

```
100747 -2875 37515 4509
77468 12111 76950 82072
-102787 41331 75324 -96228
-53535 -6367 65795 -62947
```

*独立行政法人 秋田工業高等専門学校 電気情報工学科

リスト 1: 全ての合計と各列の平均値を計算するプログラム .

```

1 #include <stdio.h>
2
3 int cal(int hoge[][4]);           // プロトタイプ宣言
4
5 double ave0, ave1, ave2, ave3;   // 各列の平均値はグローバル変数とする
6
7 //=====
8 // main 関数
9 //=====
10 int main(void)
11 {
12     FILE *in_file;
13     int i,data[10000][4], sum_data;
14
15     in_file = fopen("/home/yamamoto/tmp/program/int_data.txt", "r");
16
17     for(i=0; i<10000; i++){
18         fscanf(in_file, "%d%d%d%d",
19             &data[i][0], &data[i][1], &data[i][2], &data[i][3]);
20     }
21
22     fclose(in_file);
23
24     sum_data = cal(data);
25
26     printf("sum all = %d\n", sum_data);
27     printf("average = %f\t%f\t%f\t%f\n", ave0, ave1, ave2, ave3);
28
29     return 0;
30 }
31
32 //=====
33 // データ処理のためのユーザー定義関数
34 //=====
35 int cal(int hoge[][4]){
36     int i;
37     int sum0=0, sum1=0, sum2=0, sum3=0;
38
39     for(i=0; i<10000; i++){
40         sum0+=hoge[i][0];
41         sum1+=hoge[i][1];
42         sum2+=hoge[i][2];
43         sum3+=hoge[i][3];
44     }
45
46     ave0=sum0/10000.0;
47     ave1=sum1/10000.0;
48     ave2=sum2/10000.0;
49     ave3=sum3/10000.0;
50
51     return sum0+sum1+sum2+sum3;
52 }

```

実行結果

```

sum all = -15594426
average = -612.222100  377.635000  -430.162000  -894.693500

```

1.1.2 ファイルからのデータの読み込み

例題のリスト 1 のプログラムでは、ファイルに保存されているデータを配列に格納する方法を理解しなくてはならない。

ファイルからデータを取得するには、(1) ファイル情報を格納する変数の用意する、(2) ファイルをオープンする、(3) ファイルからデータを読み込む、(4) ファイルをクローズするという一連の動作が必要である。リスト 1 では次のようにしている。

```
FILE *in_file;

in_file = fopen("/home/yamamoto/tmp/program/int_data.txt", "r");

for(i=0; i<10000; i++){
    fscanf(in_file,"%d%d%d%d",
        &data[i][0],&data[i][1],&data[i][2],&data[i][3]);
}

fclose(in_file);
```

- `FILE *in_file;` ファイルの情報を格納する変数宣言。`*in_file` がファイルの情報を格納する変数¹と考える。先頭のアスタリスク (*) は必須である。
- `in_file = fopen("/home/yamamoto/tmp/program/int_data.txt", "r");` 読み込み用²にファイルを開いている。`int_data.txt` がファイル名で、`r` が読み込み用にファイルを開くことを示している。`fopen()` 関数の戻り値は、アスタリスクを取った `FILE` 型の変数に代入する。ここではファイルは絶対パス³で指定している。プログラムのある位置からの相対パス⁴で指定することも可能である。
- `fscanf(in_file,"%d%d%d%d", &data[i][0],&data[i][1],&data[i][2],&data[i][3]);` `fscanf()` 関数を使って、ファイルからデータを読み込む。この関数は、`scanf()` キーボードからデータを読み込む関数 `scanf()` とほとんど同じ。最初にファイル情報の変数を書いて、あとはキーボードから読み込む場合と同様にする。例で示したプログラムでは、10000 行 4 列のデータをループ文を使って繰り返して、ファイルからデータを読み込んで配列に書き込んでいる。
- `fclose(in_file);` ファイルを閉じる。使い終わったファイルは閉じなくてはならない。

1.1.3 関数への配列データの受渡し

例題のリスト 1 のプログラムで学習する第二のポイントは、ユーザー定義関数に配列のデータを渡す方法である。このプログラムでは呼び出し側では配列名のみを、ユーザー定義関数では配列名と左端を空欄とした要素数を記述している。こうすることにより、ユーザー定義関数に配列のデータを渡すことができる。

ユーザー定義関数に単純型の変数を渡す場合と配列を渡す場合では、次の点で大きく異なる。注意が必要である。

¹本当はポインター。ポインターについては教科書の 8 章

²ファイルは、書き込みと読み込みができる。

³<http://www.akita-nct.jp/yamamoto/lecture/2006/1E/2nd/html/node2.html>

⁴<http://www.akita-nct.jp/yamamoto/lecture/2006/1E/2nd/html/node2.html>

- 単純型の変数の場合，呼出側の実引数の値がユーザー定義関数の仮引数にコピーされる．したがって，ユーザー定義関数の変数の値を変化させても，呼出側の変数の値は変わらない．ようするに，別々のメモリーを使っている．
- 配列の場合，呼出側の実引数の配列とユーザー定義関数の配列は，同じメモリーを使う．したがって，ユーザー定義関数の配列の値を変化させると，呼出側の配列の値が変わる．なぜこのようにするかというと，コンピューターの資源—メモリーと CPU の計算時間—を節約したいためである．大量のデータが格納されている配列のコピーには多大な資源である．

```
sum_data = cal(data);
```

長いので省略

```
int cal(int hoge[][4]){
    ここに関数での処理の内容を書く．
    return sum0+sum1+sum2+sum3;
}
```

- `sum_data = cal(data);` 平均値と合計を計算する関数 `cal` を呼び出す．実引数は，計算するデータが格納されている配列である．配列を実引数にする場合，配列名のみを書くことに注意すべし．ここでは，`sum_data` に関数 `cal` の計算結果を格納している．
- `int cal(int hoge[][4])` ユーザー定義関数の仮引数は，配列名と要素数を書く．ただし，左端の要素数は書かなくても良い⁵．

1.2 本日の学習内容

本日はつぎに示す内容の理解を目指し，プログラムの作成の練習を行う．これは前回の授業と同一の内容であるが，さらに理解を深める．合わせて，これまで学習してきた C 言語のプログラム方法—ループ文等の制御構造や関数—の使い方の訓練も行う．

- ファイルからデータを読み込んで，配列に格納する方法を学ぶ．
- 関数に配列を渡す方法を学ぶ．

2 プログラム作成の練習

[練習 1] 次のように 20000 行 5 列，10 万個の整数型のデータがある．各列の平均値と 10 万個のデータの合計を表示するプログラムを作成せよ．データファイルは，私の講義ノートの WEB ページよりダウンロードせよ．

⁵書かなくても良い理由は，ポインターの講義で説明する

```

-46468 -73036 -74263 59744 90550
7502 75796 -71622 75060 -66033
-20720 -91952 -84563 87804 -83096
80820 -40378 -60674 -35769 -85675
77277 -14508 -42251 43698 -32137

```

この辺は長いので省略

```

35992 -15703 97768 -51670 -88513
-28625 -16030 84789 -94578 33874

```

[練習 2] 前問のプログラムに、正の整数と負の整数の数を数える機能を追加せよ。

[練習 3] 前問のプログラムに、最大値と最小値を調べ、表示する機能を追加せよ。

[練習 4] 2003 年 11 月の秋田市の毎日の 1 時間毎の気温のデータファイルがある。各行にその日の気温が 1 時間毎に書かれている。1 行目が 11 月 1 日、2 行目が 11 月 2 日、...、のようにである。以下のような表を表示するプログラムを作成せよ。気温のデータファイルは、私の講義ノートの WEB ページよりダウンロードせよ。

```

Temperature November/2003 at Akita
-----
day      max      min      average
=====
  1      20.3     6.0     12.1
  2      20.4     9.5     15.2
  3      22.8    13.9     17.8
  4      14.9     6.4     11.5
  5      18.4     4.3     10.6
  6      17.9    10.5     13.3
  :        :        :        :
  :        :        :        :
 25      10.6     7.6     9.2
 26       7.0     0.6     4.3
 27       7.3    -0.3     2.8
 28      12.2     0.1     6.0
 29      15.7     7.9    12.2
 30      15.1    11.2    13.4
-----

max(Nov.)    = 22.8
min(Nov.)    = -0.5
average(Nov.) = 9.2

```

3 課題

次の講義 (1 月 19 日) の AM8:45 までに、以下の課題をレポートとして提出すること。表紙等は、いつもの通り。表紙のタイトルは「配列 (応用 2)」とすること。

[問 1] (予) 教科書 p.248-268 を読み、以下の間に答えよ。

- 文字列定数と文字配列の違いを説明せよ。
- 文字列配列に、文字列"HELLO"を格納する方法を示せ。
- NULL 文字とは何か?

- アドレスとは何か?
- 配列名とアドレスの関係を示せ.

[問 2] (復) 以下の動作をおこなうプログラムを作成せよ.

- ファイル (/tmp/data.txt) は 1000 行 3 列, 合計 3000 個の整数が書かれている.
- ファイルを読み込んで, 最大値と最小値, そして平均値を表示するプログラムを作成せよ.