プログラミング演習 No.2

桂田 祐史

1997年10月13日

今回の目標

今回のテーマは、ベクトルを扱うプログラミング、である。

数値計算をする場合、ベクトルや行列を扱うプログラミングは非常に重要である。ところで、C 言語では、ベクトル、行列を扱うための「言語上の仕掛け」が用意されておらず、特に行列を扱うための標準的なプログラミング・スタイルと呼べるものがない (この点は Fortran と大きく異なるところである)。

筆者は、解決策のうちで、ポインターを駆使する方法を推奨している。それをブラック・ボックスとして利用するのも良いが、出来れば内容を理解して使えればそれに越したことはないであろう。そこで、今回はアルゴリズムと言うよりは、C 言語の文法理解のための問題が多い。

問題

[5] $\mathbf{R}^3$ における極座標をデカルト座標 (直交座標) に変換するコードは簡単で、

```
x = r * sin(theta) * cos(phi);
y = r * sin(theta) * sin(phi);
z = r * cos(theta);
```

とすれば良い。この逆をする (つまりデカルト座標を極座標に変換する) 関数 `d2p()` を作って、動くことをチェックせよ[1]。

```
d2p(x, y, z, &r, &theta, &phi);
```

のようにして呼び出せるようにすること。
(関数の中で値を設定する方法 (& の意味)、逆三角関数、動作チェックの方法など)

[6] 長さ n の配列で n 次元のベクトルを表現することができる。n 次元ベクトル $\vec{a} = (a_1, a_2, \cdots, a_n)$ の最大値ノルム

$$\|\vec{a}\|_\infty \overset{\text{def.}}{=} \max_{i=1,2,\cdots,n} |a_i|$$

---

[1] ちょっと短かすぎて悪い名前だが、Decartes 座標 to Polar 座標、のニュアンス。

を計算する関数を作ることを目標にする。手始めに、標準入力から入力された3次元ベクトルの最大値ノルムを計算する、次のようなプログラムを書き始めた。関数 `maxnorm0()` を完成させよ。

```
#include <stdio.h>

#define N 3

main()
{
    int i;
    double a[N], maxnorm0();
    for (i = 0; i < N; i++)
        scanf("%lf", &a[i]);
    printf("maxnorm=%g\n", maxnorm0(a));
}

double maxnorm0(....
```

(関数に配列を引数として渡す方法、最大値の計算法など)

[7] (前問の続き) ベクトルの長さ n は、プログラムの実行時に変更したいと考えて、次のようなプログラムを書き始めた。関数 `maxnorm()` を完成させよ。

```
#include <stdio.h>

#define MAXN 1000

main()
{
    int i, n;
    double a[MAXN], maxnorm();
    printf("input n (<= %d): ", MAXN); scanf("%d", &n);
    if (n > MAXN)
        exit(1);
    for (i = 0; i < n; i++)
        scanf("%lf", &a[i]);
    printf("maxnorm=%g\n", maxnorm(a, n));
}

double maxnorm(....
```

できれば `maxnorm()` の定義は別のファイルに書いて、分割コンパイルできるようにする。
(再利用可能性の高い関数を書く方法など)

[8] 前問のプログラムでは `MAXN` の定義を書き換えれば、大きな次元のベクトルに対応するプログラムに変更できるが、それでもプログラムに変更が必要なのは面倒である。また、大きな `MAXN` をあまり大きくすると、無駄である。そこで、ベクトルの長さ n をプログラムの実行時に指定できるようなプログラムが望ましい。これは次のように `malloc()` を使うように書き換えれば良い。

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

main()
{
    int i, n;
    double *a, maxnorm();

    printf("input n: "); scanf("%d", &n);
    a = (double *)malloc(n * sizeof(double));
    if (a == NULL) {
        fprintf(stderr, "メモリーが確保できません\n");
        exit(1);
    }
    /* 以下、プログラムの字面は前問とまったく同じで OK */
    for (i = 0; i < n; i++)
        scanf("%lf", &a[i]);
    printf("maxnorm=%g\n", maxnorm(a, n));
}

double maxnorm(....
```

(C++ を勉強しているならば、同じことを C++ で書いてみよ。)
長さ n のベクトルを標準入力から二つ読み込んで、それらの和、内積、ユークリッド・ノルムと、二つのベクトルのなす角度を計算するプログラムを作れ。後で、利用できるように、`addvector()`, `dotproduct()`, `norm()`, `angle()` のような関数を作ること。

[9] ある理由から

`a[0]`, `a[1]`, `a[2]`, $\cdots$, `a[100]`

の両端 `a[0]`, `a[100]` を除いた

`a[1]`, `a[2]`, $\cdots$, `a[99]`

の絶対値の最大値を知りたいとする。`maxnorm()` を使って求めるには、どうすればよいか。