

ソートとシャッフル

カードゲームをするときなど、作品を作る上で「シャッフル」をしなければならない機会は数多く存在します。

今回は、この「シャッフルする」という処理を、「ソート（並び替え）」を用いて再現してみたいと思います。

○ソート（並べ替え）について…

例えば、「7, 3, 11, 2, 13, 5」とでたらめに並んでいる数字を昇順に並べる（小さい値から大きい値に並べる）にはどうすれば良いでしょうか？

その方法として、前後の数値の大小比較を行って並べ替えるバブルソート法という方法があります。バブルソート法プログラムの手順は下記のようになります。

- ① 与えられた数値全てを、順に変数に代入します。
- ② 先頭の数値とその次の数値の大小を比較し、先頭の数値が次のより大きいのであれば互いに位置を入れ換えます。この場合だと、「7」と「3」が比較されるので互いの位置が入れ換えられます。
- ③ 一つずらした前後の数値の大小を比較し、大きいのであれば互いに位置を入れ換えます。この場合だと、「7」と「11」が比較されるので入れ換えは起きません。この操作を繰り返していくと、並びは「3, 7, 2, 11, 5, 13」となり、この並びの中で一番大きな数値の「13」が最後尾に来ます。
- ④ また先頭の数値に戻って②と③の操作をする。このとき、最後尾の数値は既に一番大きな数値になっているので、その一つ手前のところで大小比較を終える様にしましょう。

こうして、でたらめに並んでいた数値は「2, 3, 5, 7, 11, 13」と昇順に並び直されます。ソート法は他にも、選択ソート法やクイックソート法など、処理を軽くするために色々な方法が考案されています。

バブルソート法では配列の変数を宣言し、与えられた数値を全て順に代入した後、二重 for 文を用いて比較をしていきます。個々の値の大小比較の際は if 文を用います。このとき、値を入れ換えるときは、変数（例「a[0]」）から変数（例「a[1]」）に直接代入して入れ換えず、中継役の変数（例「temp」）を介して入れ換えるようにしましょう。

では、「1, 2, 3, 4, 5, 6, 7」と昇順に並んでいるものをでたらめに入れ換えればどうなるのでしょうか？もちろん、並び方はでたらめになります。シャッフルとは、このようにして再現することが出来ます。

○でたらめをもたらす乱数について…

C++ Builder には乱数を生成するために `random()` という関数があります。この発展編の冊子でも乱数について触れた章があります。この関数で作られた乱数は、プログラムを何度も実行するといくらか偏りの生じた結果となりがちです。

今回は別の関数、`srand()` を用いて乱数の分布をある程度均一になるようにしたいと思います。この `srand()` 関数を使うためには、乱数生成のため新たな種 (seed) を用意しなければなりません。

この種に当たるものは色々用意できます。今回は `<time.h>` ライブラリにある `time(NULL)` を用いて乱数の種にします。これは、1970年1月1日0時0分0秒から現在までの時間経過を秒単位で出力するという機能があります。

ヘッダーファイルにて `<time.h>` ライブラリを追加した後、下記のように書いて乱数を再設定してみましょう。

```
srand((unsigned)time(NULL));
```

ここで、()内にある (unsigned) とは、キャストのことです。これで `time(NULL)` の中身を正の整数にしています。

これで乱数生成の準備が出来ました。後は rand() という関数を使って乱数を生成し、前もって用意した変数に代入しましょう。この変数を用いてソートの入れ換え手順を行えば、シャッフル機能の実現です！

```
//-----  
void __fastcall TForm1::Shuffle()  
{  
    //ソートによるシャッフル  
    int trump_n, temp, change_num;  
    /*  
        「trump_n」はトランプの総枚数（52枚）。  
        「trump[]」は個々のトランプを示す番号（52個）。  
        ヘッダーにて宣言。  
        「change_num」は入れ換え場所。  
        「temp」は中継役。  
    */  
  
    srand((unsigned)time(NULL)); //乱数の種生成  
  
    for(int i = 0; i < trump_n; i++){  
        change_num = rand() % trump_n; //入れ換える場所の決定  
        temp = trump[i];  
        trump[i] = trump[change_num];  
        trump[change_num] = temp;  
    }  
  
}  
//-----
```