Cプログラミング演習1(再)
4

講義では、Cプログラミングの基本を学び

演習では、やや実践的なプログラミングを通して学ぶ

今回のプログラミングの課題

(前回の課題で取り上げた）data.txt の要素をソートして、
sorted.txt というファイルに書出す

ソート(sort) とは：

　数の場合、小さいものから大きなもの(昇順）、もしくは
大きなものから小さなもの(降順）、になるよう、
並び替えること

　文字列の場合は、「辞書式順」(アルファベット順、もしくは「あいうえお順」)が使われる

プログラムが作れたら…

data.txtの中身

```
2
3
-5
0
9
12
-10
8
-3
9
```

ソートの結果:

```
-10
-3
-5
0
2
3
8
9
9
12
```

プログラムの手順

(1) 入力:　data.txt

(2) 処理: ソート（いろいろな方法がある）

(3) 出力: ソートした結果を画面表示する(ファイルに書出す）

プログラムの手順

(1) 入力:　data.txt

　　ファイルから要素を読み込み、配列に記憶する

(2) 処理: ソート（いろいろな方法がある）

(3) 出力: ソートした結果を画面表示する(ファイルに書出す）

プログラムの手順

(1) 入力:　data.txt

ファイルから要素を読み込み、配列に記憶する

適当な大きさの配列を用意する
ファイルの読み込み準備（ファイル変数）
ファイルから要素を一個ずつ読み込んで、
　配列に記憶する

(2) 処理: ソート（いろいろな方法がある）

(3) 出力: ソートした結果を画面表示する(ファイルに書出す）

プログラムの手順

(1) 入力:　data.txt

注意：配列で1番目の要素のインデックスは 0
同様に 2番目の要素のインデックスは 1

　ファイルから要素を読み込み、配列に記憶する

(2) 処理: ソート（いろいろな方法がある）

　ここでは、昇順の選択ソート(selection sort)を考えよう

　「配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換する。(これにより1番目の要素が「最小」となる)

　次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換する。(これにより2番目の要素が「2番目に最小」となる)

　…　この操作を配列の最後の要素まで繰り返す」

(3) 出力: ソートした結果を画面表示する(ファイルに書出す)

昇順の選択ソートプログラム

「配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換する。
次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換する。
…　この操作を配列の最後の要素まで繰り返す」

ちょっと難しそう。。。なにをどうしたらよいのか？

注意：配列で1番目の要素のインデックスは 0
同様に 2番目の要素のインデックスは 1

考え方：「何かを繰り返す」と書いてあるのだから、
繰り返される処理を「関数」として捉える

昇順の選択ソートプログラム

注意：配列で1番目の要素のインデックスは０

配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換

次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換…

この操作を配列の最後の要素まで繰り返す

これに注目

繰り返される処理を「関数」として捉える

青字：入力
茶色：処理

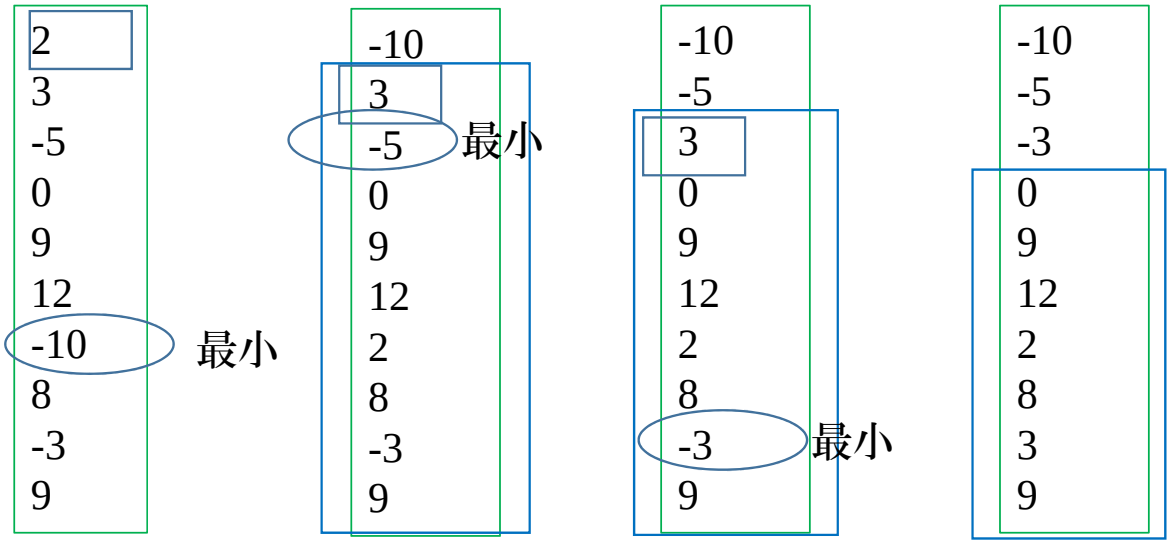
配列のi番目以降の要素で最小の要素を探し、ｉ番目の要素と交換する

最終的に得られるのは… 要素が交換された配列

確認：この部分でも当てはまる？

昇順の選択ソート

「配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換。
　次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換…　この操作を配列の最後の要素まで繰り返す」
という方法が実際にうまくいくのか、自分の手で確かめてみる:



注意：配列で1番目の要素のインデックスは 0
2番目の要素のインデックスは 1
…

うまくいきそう。。。

昇順の選択ソートプログラム作成

「配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換。
　次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換…　この操作を配列の最後の要素まで繰り返す」

まず、繰り返される処理を行う関数の名前、入力、処理、出力を定める

関数名を、（仮に）findAndReplace とする

入力：　整数の配列　int arr[ ]
　　整数　int i, int n

配列 arr の要素数を nとする

処理: i番目以降の要素のうちの
　最小値を見つけて
　i番目の要素と取り替える

出力：　書き換えられた整数の配列　int arr[ ]

あとで修正

昇順の選択ソートプログラム作成

「配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換。
　次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換…　この操作を配列の最後の要素まで繰り返す」

処理: i番目以降の要素のうちの
　最小値を見つけて
　i番目の要素と取り替える

注意：配列で1番目の要素のインデックスは 0
2番目の要素のインデックスは 1
…

やるべき仕事は２つ
(1) 配列のi番目以降の要素から最小値を見つける
(2) その要素とi番目の要素を入れ替える

「配列から最小値を求めるプログラム」の改訂

配列の1番目の要素（インデックス0）から最後の要素(インデックスn-1)までの要素のうちの最小値を求めるプログラム

… は前に作った

今回はこれを元に

注意：配列で1番目の要素のインデックスは 0

配列に対し、インデックスiからn-1までの要素のうちの最小値と最小値の要素の番号(インデックス)を求める」関数

を作る

そして、最小値とインデックスi番目の要素を入れ替える…

そのために、最小値の要素のインデックスも記憶する！

「配列から最小値を求めるプログラム」の改訂

第２回目演習で「配列の要素の最小値を求める」関数を作った:

これを元に「インデックスiからn-1までの要素のうちの最小値とその番号(インデックス)を求める」関数にするには、以下をどのように変更するかを考えれば良い：

(1)入力
(2)処理
(3)出力

注意：2つのもの（ここでは最小値とそのインデックス）を返す方法はまだ学んでいない…
しかし！　インデックスさえわかれば最小値がなんであるかは分かる！
そこで、「最小値のインデックス」を求める関数を考えてみよう

関数 minInArray

出力＝記憶した最小要素 は int 型

```
int   minInArray (int ar[] , int n)
{
    int min=ar[0];
    int i;
    for (i=1; i< n; i++)  {
      if (min > ar[i]) // 一個ずつmin値と比較
            min = ar[i];   // minxよりも大きければ更新
    }
    return  min;
}
```

入力＝整数を要素とする配列
int ar[]　と 要素数 int n

今まで見た要素の中で最小のものを記憶する」変数を宣言、その変数に初期値を与える

要素を比較して最小値を求める

出力=最小要素

「配列から最小値を求めるプログラム」の改訂

配列に対し、インデックスiからn-1までの要素のうちの最小値と
そのインデックスとを求める」関数に作り変えてみよう

(1)入力

(2)処理

(3)出力

昇順の選択ソートプログラム作成

「配列の1番目以降の要素で最小の要素を探し、1番目の要素と交換。

次に配列の2番目以降の要素で最小の要素を探し、2番目の要素と交換…　この操作を配列の最後の要素まで繰り返す」

処理: インデックス i 以降の要素から
(1) 最小値の要素のインデックスを見つけて
(2)インデックス i の要素と取り替える

(1) 今作った関数の名前を findMinValueとする。
findMinValue(arr, i n)によりインデックスi番目からn番目までの要素のうち、最小値要素のインデックスを返す

(2) findMinValue(arr, i, n)の値を k とすると、 arr[i]とarr[k]の値を取り替える(swap)

配列の要素の値の取替え（ちょっと落とし穴）

「arr[i]とarr[k]の値を取り替える(swap) 」についての注意

これは以下のようにすればできる、と勘違いする人が時々いる…

arr[i] = arr[k]

arr[k] = arr[i]

これではできない！

例: arr[i]=10, arr[k]=3とおくと、最初の実行で arr[i]=arr[k]=3 となり、 arr[i]が持っていた値 10が消えてしまうからである！

解決策： 変数をもう一つ用意する(tempとする）
そして、 temp = arr[i] ; arr[i]=ar[k]; arr[k]=temp
とする（これが動くことを確かめよ）

コツ：変数をケチらないこと

関数 findAndReplace の完成

関数 findMinValue を使う

void となっているのは、
「書き換えられた配列」を
内部で作っているから、
配列を出力しなくてよいため

```
void findAndReplace(int arr[ ], int i, int n) {

}
```

昇順の選択ソートプログラムの作成

整数を要素とする配列 arr 、そのインデックス0からn-1まで(配列の要素数はn)の要素をソートする:

```
// arrには要素が既に入っているとする
for (i=0; i < n; i++) {
        findAndReplace(arr, i, n-1);
}
// この結果 arr の中身はどうなっているだろうか？
```

全体のプログラムの完成

data.txt の要素をソートして、sorted.txt というファイルに書出す

```
#include <stdio.h>
// 関数 findMinValue を定義
// 関数 findAndReplaceを定義
int main(void) {
	// 必要な変数の定義
	// ファイルdata.txtから要素を読み込み 配列 arr  に記憶
	//　　そのとき、読み込んだ要素数も記憶 (変数nの値とする)
	//  選択ソートにより、配列 arr の中身を昇順ソート
	// arr の中身を sotred.txt に書出す
	return 0;
}
```

課題の提出

プログラムを完成させ、
適当なファイル(例えばdata.txt)を与えて
その要素をソートし
その結果がsorted.txtファイルに書き込まれる

ことを確認しよう

できたらプログラム（説明付き）と実行結果を提出する。
　　プログラムは適切にインデントすることをお忘れなく