Problem Set 3 Exercise #17: Sort Three Digits

Reference: Lecture 8 notes

Learning objective: Sorting

Estimated completion time: 30 minutes

Problem statement:

We have illustrated Selection Sort algorithm using an integer array in class. Determining whether one element, say **`arr[i]`**, is greater than its another, say **`arr[j]`**, is simply done by comparing them (e.g. **`if (arr[i] < arr[j])`**).

What if the array elements are more complex (for example, each element is a string, or is a structure comprising of more than one component — we will cover string and structure later), or the comparison criterion is more complex?

Suppose you want to sort an integer array in increasing order of the first 3 digits of each element, how would you modify the Selection Sort algorithm given in the lecture notes?

For example, in the sample run #1 below, the first 3 digits of the 6 values are 123, 987, 32, 555, 801 and 729. These 6 values are thus sorted and printed out in ascending order of their first 3 digits.

You may assume there are at most 100 integers in the array.

Write a program **sort_three_digits.c** for the above task.

Sample run #1:

```
Enter the number of elements: 6
Enter 6 elements: 12345 9870 32 5555555 801784 729
Sorted array: [32, 12345, 5555555, 729, 801784, 9870]
```

Sample run #2:

```
Enter the number of elements: 5
Enter 5 elements: 23456 3456 456 56 6
Sorted array: [6, 56, 23456, 3456, 456]
```