

Practice Examples for Lab: Set 13

- 1

Suppose you are given a sequence of numbers, preceded by the length of the sequence. You are required to sort them. In this exercise you will do this using the so called *Insertion sort* algorithm. The idea of the algorithm is to read the numbers into an array, but keep the array sorted as you read. In other words, after you read the first i numbers, you must make sure that they appear in the first i elements of the array in sorted (say non-increasing) order. So when you read the $i + 1$ th number, you must find where it should be inserted. Suppose you discover that it needs to be placed between the numbers that are currently at the j th and $j + 1$ th position, then you should move the numbers in positions $j + 1$ through $i - 1$ (note that the indices or positions start at 0) forward in the array by 1 step. Then the newly read number can be placed in the $j + 1$ th position. Write the program that does this.

- 2

Write a program that reads a sequence of names, one per line, and then sorts and prints them.

Practice Examples for Lab: Set 13

• 3

Write and test the following function that attempts to remove an item from an array:

```
bool removeFirst(float a[], int& n, float x);
```

The function searches the first `n` elements of the array `a` for the item `x`. If `x` is found, its first occurrence is removed, all the elements above that position are shifted down, `n` is decremented, and `true` is returned to indicate a successful removal. If `x` is not found, the array is left unchanged and `false` is returned.

• 4

Write and test the following function:

```
void rotate(int a[], int n, int k);
```

The function “rotates” the first `n` elements of the array `a`, `k` positions to the right (or `-k` positions to the left if `k` is negative). The last `k` elements are “wrapped” around to the beginning of the array. For example, the call `rotate(a, 8, 3)` would transform the array `{22,33,44,55,66,77,88,99}` into `{77,88,99,22,33,44,55,66}`. The call `rotate(a, 8, -5)` would have the same effect.

Practice Examples for Lab: Set 13

- 5

Rewrite bubble-sort as an indirect sort, i.e., instead of moving the actual elements of the array, sort an “index” array.

- 6

Write and test the following function:

```
void reverse(int a[], int n);
```

The function reverses the first n elements of the array. For example, the call `reverse(a, 5)` would transform the array `{22,33,44,55,66,77,88,99}` into `{66,55,44,33,22,77,88,99}`.

- 7

Write and test a function that implements the *Perfect Shuffle* of a one-dimensional array with an even number of elements. For example, it would replace the array `{11, 22, 33, 44, 55, 66, 77, 88}` with the array `{11, 55, 22, 66, 33, 77, 44, 88}`.