ENEE 140 Week 12

This handout includes instructions for the recitation sessions on Wednesday and Friday.

Follow these instructions to review arrays and sorting operations, then **submit the homework** as indicated below.

1 Array removal and insertion

Given an integer array `a[8] = {2, 3, 5, 7, 11, 13, 17, 19}`, read three integers `k`, `m`, and `x` from the terminal. When `k` is a valid array index (i.e., between 0 and 7), remove `a[k]` from the array by shifting all the elements from `k+1` to the end of the array one place to the left (so that `a[k]` takes the value of the old `a[k+1]`, `a[k+1]` takes the value of the old `a[k+2]`, etc.). Then, if `m` is a valid array index, insert integer `x` on position `m` of the array and shift the elements from `m+1` to the end of the array one place to the right (so that `a[m+1]` takes the value of the old `a[m]`, `a[m+2]` takes the value of the old `a[m+1]`, etc.).

After these two operations, print out the elements in the array. The array should not have lost any elements other than the old `a[]`, and all the old elements should have retained their original order.

For example, if we have `k`=2, `m`=5, `x`=10 then the array will be changed to `{2, 3, 7, 11, 10, 13, 17, 19}`.

Homework

Due: Friday at 11:59 pm.

Create a program by following the instructions below.

Log into Elms, click on Gradescope in the course menu, then go to the relevant assignment to submit your work.

2 Insertion sort

Write a complete program, called `insertion_sort.c`, that sorts an array of integers. Insertion sort is similar to the selection sort algorithm in that, after the `k`-th iteration, the first `k` elements in the array are in sorted order. To sort the array `a[]` in ascending order (smallest to the largest), insertion sort works as follows:

- In iteration `k`, copy `a[k]` to a temporary variable, then insert this variable into the array region between `0` and `k-1` (which is sorted) on its correct position in the sorted order. You will have to shift the elements between that position and k one place to the right.
- Start from `k`=1, as the array region before k is trivially sorted (it has only one element).
- Before implementing the function, apply insertion sort manually to sort the following array, in order to visualize how this algorithm works.
  `{20, 5, 11, -3, 0, 34, -12, 9, 100, 4}`

3 Comparison of the sorting algorithms (optional)

Think about the two sorting algorithms we have learned by this point, selection sort and insertion sort. How many comparisons does each algorithm perform? You do not have to submit the answer to this question.

Reading assignment

Review the material for the final exam.

Weekly challenge

No challenge this week.