CSE 3318-003 Lab Assignment 1

Due February 5

Goal:

Understanding of sorting concepts.

Requirements:

1. Write a C program to take a sequence of `n` integer values and remove duplicate occurrences, i.e. only the first occurrence of each value will remain. `n` will be the first input value and the sequence may be read using `scanf()`s. The phases for performing this task are:

   a. Read input sequence of values, each value giving an ordered pair (value, position indicator).
   b. Sort the pairs in ascending order by value in a stable fashion. If your chosen sort is stable, then the key is just the value. If your chosen sort is unstable, the key must be extended with the position indicator.
   c. Using one pass ($\Theta(n)$ time) over the sorted result, remove any occurrences beyond the first one for a key.
   d. Sort the pairs using the (unique) position indicator as the key. (Stability is not an issue.)
   e. Output the number of unique values followed by the values (without the position indicators).

   The input will be read from standard input (`stdin`) as either keyboard typing or as a shell redirect (<) from a file. Prompts/menus are completely unnecessary!

2. Submit your program on Canvas by 3:45 p.m. on Wednesday, February 5. One of the comment lines should indicate the compilation command used on OMEGA.

Getting Started:

1. You may use any sort you wish, but the standard library `qsort()` is highly recommended.

2. Processing for the input:

Input:	a. Array of pairs	b. After sorting	c. Remove extras	d. After sorting	e. Output:
10	0: 3 0	0: 1 5	0: 1 5	0: 3 0	7
3	1: 3 1	1: 2 4	1: 2 4	1: 7 2	3
3	2: 7 2	2: 2 8	2: 3 0	2: 5 3	7
7	3: 5 3	3: 3 0	3: 4 6	3: 2 4	5
5	4: 2 4	4: 3 1	4: 5 3	4: 1 5	2
2	5: 1 5	5: 3 7	5: 7 2	5: 4 6	1
1	6: 4 6	6: 4 6	6: 9 9	6: 9 9	4
4	7: 3 7	7: 5 3			9
3	8: 2 8	8: 7 2			
2	9: 9 9	9: 9 9			
9					

3. Do not include intermediate results (as in 2.) in the output for the version you submit on Canvas.