Sorting Algorithms Continued Solutions

partial_sort()

- Briefly describe the std::partial_sort() function
  - partial_sort() takes an iterator to an element in the range
  - The offset of this iterator gives the number of elements which will be sorted in the result
  - The remaining elements will come after the sorted elements, but will not be ordered relative to each other
  - e.g. if the iterator is begin() + 5, the first 5 elements in the result will be the "top 5" elements, in order
  - The remaining elements can be in any order

partial_sort_copy()

- Briefly describe the std::partial_sort() function
  - partial_sort_copy() does a partial sort and stores the result in a destination
  - It sorts as many elements as will fit into "dest" and writes them there
  - If the destination is large enough, it will sort the entire range

nth_element()

- Briefly describe the std:: nth_element() function
  - nth_element() takes an iterator to an element in the range
  - It rearranges the elements so that the iterator points to the element that would be in that position if the range was sorted
  - e.g. if the iterator is begin() + 4
  - The iterator will point to the element which ranks 4th
  - It then performs a partition with this element as the partition point
  - All the elements before it will have a lower value
  - All the elements after it will not have a lower value