Choose **one** scribe from each group. Please turn in a copy at the end of class.

Name: _____, _____

Name: _____, _____

1. Suppose the query "Colgate AND Raiders" is submitted to our search engine. We want to design an algorithm that finds all documents that contain *both* the word "Colgate" and the word "Raiders."

   Your tasks:

   - Design an *efficient* algorithm. Write it down as precisely as possible. If you want to make additional assumptions about the structure of the index, you may but please describe them. Hint: use binary search and sorting as useful *building blocks*.
   - Analyze its runtime. You can assume the following: the word array has m words. The entry for "Colgate" has n_1 words and the entry for "Raiders" has n_2 words

   For an additional challenge, think about how your algorithm would change if...

   - You wanted to support queries with more than two words, such as "Colgate AND Raiders AND Track AND Field."
   - You wanted to support *phrase* matches. E.g., the query "Colgate Raiders" should return pages where "Colgate" and "Raiders" appear next to each other in the document.