General Clarifications:

1. I/O formats to be exactly the same as homework 2/3.
2. Free text searches embedded within Boolean retrieval terms are supposedly possible

Basic Functionality:

* Parsing
  + Find a consistent delimiter to split the database up. Get structure as well to see how we can decide between fields and zones
* Indexing
  + How are fields handled? E.g. if you search for a lawyer’s name or judge’s name  
    1 posting-dict pair or more?
* Boolean retrieval and free text search
  + Functionality and technical implementation need to be discussed. Determines how a query is processed.
  + Handle terms in the Boolean retrieval like free text or like phrases? Bag of words vs bigrams.
* Boolean retrieval
  + Skip pointers
  + Shunting yard query processing

Additional Features:

* Stop-word removal
* Modified shunting yard for mass intersection
* NLTK thesaurus
* Tier-based scoring
  + Add points for more than 1 term matches in the query
  + Add points for matching fields instead of zones (req indexing)
* Degrees of separation
  + Cases refer to other cases. Include other cases but diminish relevance according to degrees of separation
  + Zero-th degree could be top 3 results or documents referenced in the query.