1.

	Worst Time Complexity	Best Time Complexity
addPostToDatabase(User u, Post p)	O(n)	O(1)
computeMostUrgentQuestion()	O(n)	O(n)
retrievePost(User u)	O(n)	O(n)
deletePostFromDatabase(User u, Post p)	O(n)	O(1)

Best case scenario:

addPostToDatabase: The best case occurs when the user is at the first of the user list so it does not need to iterate through the user list to see whether the user is enrolled.

computeMostUrgentQuestion: The best scenario is the same as the worst because it always needs to iterate through the array to compare the priority.

retrievePost: The best scenario is the same as the worst because it always needs to iterate through the array to find whether the post was associated with the user.

deletePostFromDatabase: The best scenario is when the deleted post is at the end of the array so that the array does not need to shift the elements, and also the user is at the first of the user list so that it does not need to iterate through the user list.

2. Advantages: 1. Arraylist is not limited to the length.
   2. It is easy to insert or remove the particular element in the arraylist.
   3. It is easy to get the element by index.
   4. We can easily access the elements in the arraylist.

   Disadvantages: 1. When the element is added to the arraylist or removed from the arraylist, if it is not at the end of the arraylist, the arraylist needs to shift the elements inside.
   2. Once the arraylist exceeds its capacity, it has to copy the current array and creat a new array.

3. Throughout theses questions, I changed the data structure Arraylist to Hashtable. Because compared to the Arraylist, Hashtable has a lower average runtime of O(1). So when I am adding a new post, a hashtable just need to find the hash value and insert the keyword as the key and the Post arraylist as the value. It saves a lot of time for inserting, deleting, and searching, because I just need to use the key to find the node and I will get a list of post with the same keyword. I built three hashtables. The first is keyword hashtable which takes keyword as the key, and post arraylist as the value. Second is user hashtable which takes user as the key, and post arraylist as the value. Third is unanswered hashtable which takes UID as the key, and the post as the value. Therefore, it will be quick and easy to access the target post.