|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | ResizeableArrayBag |  |  | LinkedBag |  |
|  | Union | Intersection | Difference | Union | Intersection | Difference |
| Time Complexity in the Best Case | O (1) | O (1) | O (1) | O (1) | O (1) | O (1) |
| Time Complexity in the Worst Case | O (n) | O (n^2) | O (n^2) | O (n) | O (n^2) | O (n^2) |
| Explanations | In the best case, one or both bags are empty, and the non-empty bag, or whichever empty bag isn’t checked first, is returned instead of combining an empty bag. In the worst case, both bags have elements, and the method must traverse each bag, adding the elements to a new bag. | In the best case, one or both bags are empty. If one or both bags are empty, there can be no overlap of elements, so the resulting intersection is an empty bag. In the worst case, both bags contain elements and must be traversed through and checked for overlap. The contains method must be used in the worst case so that is a for loop within a for loop and therefore it is n^2 because of the nested loops. | In the best case, one of the bags is empty. If the first bag is empty, then the difference between the first bag and an empty bag is the first bag. If the second bag is empty, then the difference between an empty bag and a non-empty bag is an empty bag. In the worst case, both bags are not empty and must be traversed to find the difference between each. The worst case is n^2 because we have nested loops to check for the difference between bags. | In the best case, one or both bags are empty, and the non-empty bag, or whichever empty bag isn’t checked first, is returned instead of combining an empty bag. In the worst case, both bags have elements, and the method must traverse each bag, adding the elements to a new bag. | In the best case, one or both bags are empty. If one or both bags are empty, there can be no overlap of elements, so the resulting intersection is an empty bag. In the worst case, both bags contain elements and must be traversed through and checked for overlap. The contains method must be used in the worst case so that is a for loop within a for loop and therefore it is n^2 because of the nested loops. | In the best case, one of the bags is empty. If the first bag is empty, then the difference between the first bag and an empty bag is the first bag. If the second bag is empty, then the difference between an empty bag and a non-empty bag is an empty bag. In the worst case, both bags are not empty and must be traversed to find the difference between each. The worst case is n^2 because we have nested loops to check for the difference between bags. |

Efficiency