

1. hashing. $O(n)$ expected.
2. reduce to integer sorting. after sorting, only need $O(n)$ time by monotone pointers.
for integer sorting:
 $O(n\sqrt{\log \log n})$ randomized [2].
 $O(n \log \log n)$ deterministic [3].
3. for the lower bound, $\Omega(n \log n)$ in the degree- d algebraic decision tree model [1]. we can also reduce the set intersection problem to this, i.e. deciding whether two sets $\{x_1, \dots, x_n\}$ and $\{y_1, \dots, y_n\}$ intersect, by negating one of the sets.

References

- [1] Alexander C. L. Chan, William I. Gasarch, and Clyde P. Kruskal. Refined upper and lower bounds for 2-sum. 2004.
- [2] Y Han and M Thorup. Sorting integers in $O(n \log \log n)$ expected time and linear space. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2002.
- [3] Yijie Han. Deterministic sorting in $O(n \log \log n)$ time and linear space. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 602–608. ACM, 2002.