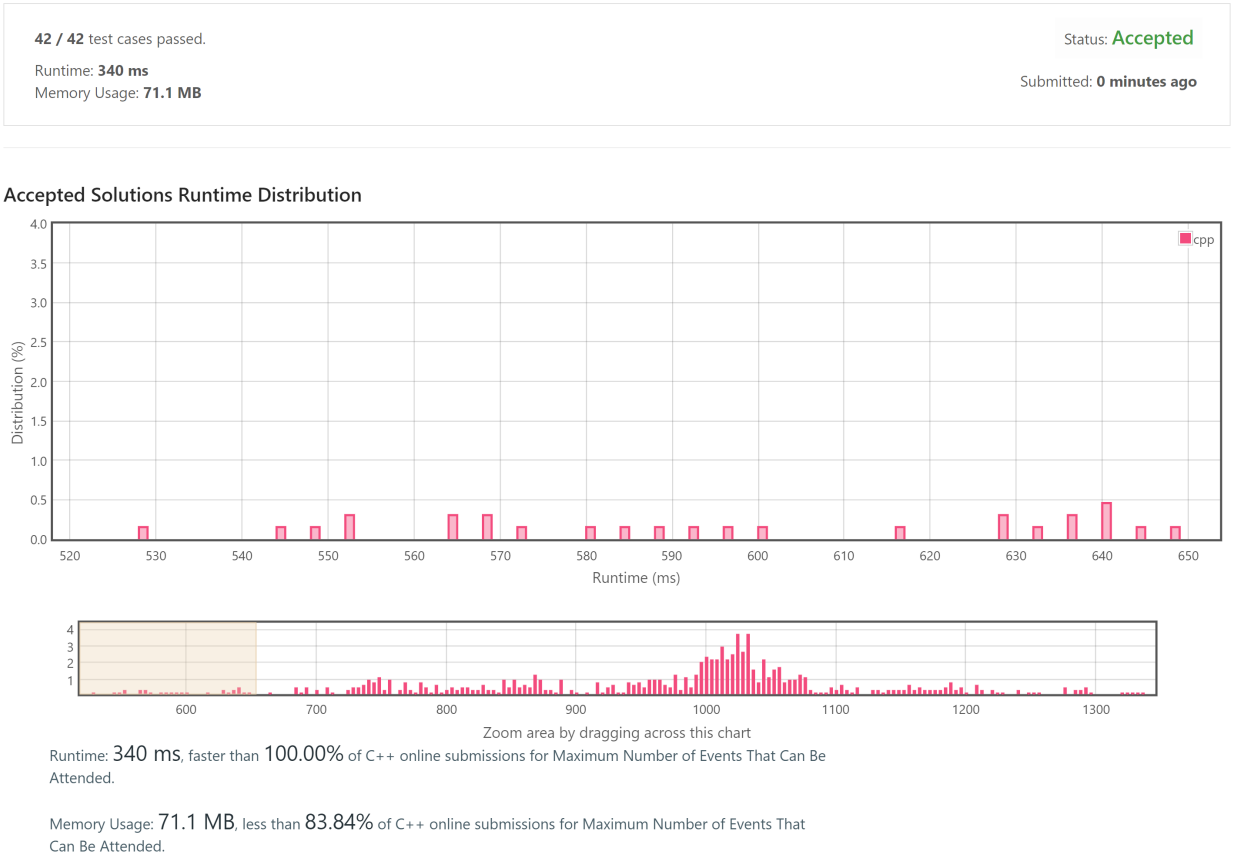
1. Greedy, heap. $O(n \log n)$.
2. Sorting according to the right endpoint in increasing order, then for each interval, greedily place at the leftmost possible position. Two endpoints with distance at least n cannot affect each other, so we can wlog assume $U = O(n^2)$, and thus sorting takes $O(n)$ time. Union find can be implemented in $O(n)$ time in this case (deletion only, using bit packing) [1]. We can also avoid union find, by scanning from left to right. $O(n)$.



References

[1] Harold N Gabow and Robert Endre Tarjan. A linear-time algorithm for a special case of disjoint set union. *Journal of computer and system sciences*, 30(2):209–221, 1985.