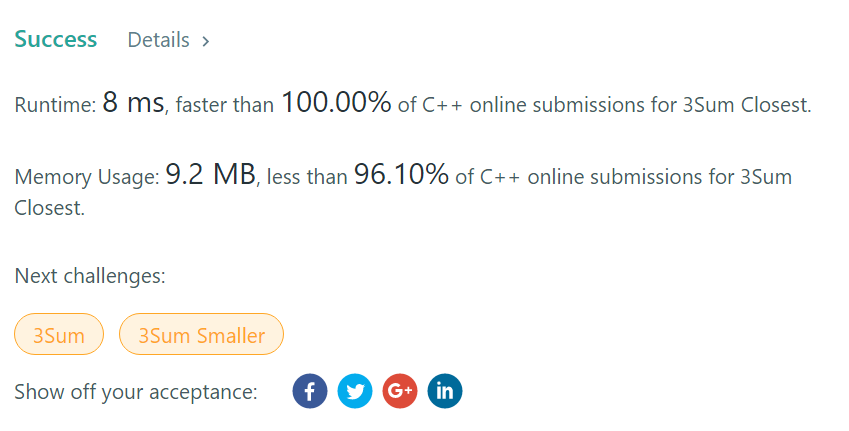
第一个版本：

利用15题的固定一个元素i，然后对剩余元素执行双向扫描方法（双向推进），根据情况对两个边界进行调整，找到两个元素lo和hi。

1. 利用双向逼近可能更快，因为两边的相互抵消（前为负数，后为大数）。
2. MinDist == 0 时退出查找，加速查找
3. 顺序访问，对访存友好（缓存友好），而第二个版本是二分查找，理论上会更快，然而二分是跳跃式查找，对访存不友好，因此实际性能会下降。



第二个版本：

固定两个元素i 和 j，然后对剩余元素执行二分法查找，找到一个元素mid。

充分利用零点思想，划分四种情况。

