

برای حل این بخش به دلیل زیادبودن فاصله بین کوچک‌ترین و بزرگ‌ترین عددهای لیست، استفاده از روش قسمت قبل کاربردی نخواهد بود و با استفاده از جدول درهم سازی نوشته شده در سوال ۴ می‌توانیم به صورت تقریبی به کارایی مورد نظر دست یابیم. کارایی زمانی دسترسی به عناصر با استفاده از جدول درهم‌سازی در بدترین حالت یعنی زمانی که تعداد زیادی از کلیدها hash یکسان داشته باشند می‌تواند $O(n)$ باشد اما به صورت میانگین از آنجایی که اندازه جدول چهار برابر تعداد عناصر انتخاب شده است، هزینه دسترسی به عناصر جدول $O(1)$ خواهد بود و می‌توانیم جواب قسمت ب را در $O(n)$ پیدا کنیم.

```
def find_a_b(x: int, A: list[int], B: list[int]):
    n = len(A)
    hash_table = hash_table_module.HashTable(4 * n)
    for number in A:
        hash_table.add(number)
    for number in B:
        if hash_table.search(x - number):
            return True
    return False
```