$O(n)$ time.

1. (in-place) reverse half of the list. only need additional $O(1)$ space, but the space complexity is $O(n)$.

2. randomized $O(1)$ space, correct w.h.p.:

like Rabin-Karp, use string hashing. Let s be a string of length n, $h(s) = (\sum_{i=0}^{n-1} s_i x^i) \mod p$, where p is a prime arbitrarily selected within $[n^2, 2n^2]$, and x is selected uniformly at random from $\{0, \ldots, p-1\}$. For two strings $s \neq t$, $\Pr[h(s) = g(t)] \leq \frac{1}{n}$. We can compute the hash function incrementally either left to right or right to left, each step we need to either multiply x, or multiply the inverse of $x \mod p$. finding a prime p and computing the inverse of x takes $O(1)$ space.

lower bound: assume the linked list is placed on a read-only tape. there is no deterministic $O(n)$ time and $O(1)$ space algorithm?

assume we only have $O(1)$ space, therefore at any time, we only have access to $k = O(1)$ positions in the linked list. in one step we can move one position to the right, delete the access to one position, or copy one position.?

References