```c
struct ListNode* reverseList(struct ListNode* cur, struct ListNode* prev) {
    while (cur != NULL) {
        struct ListNode* temp = cur->next;
        cur->next = prev;
        prev = cur;
        cur = temp;
    }
    return prev;
}

// Main function to calculate the maximum sum of pairs.
int pairSum(struct ListNode* head) {
    int res = 0;
    struct ListNode* slow = head;
    struct ListNode* fast = head;

    // Finding the midpoint of the list using the Floyd's tortoise and hare algorithm.
    while (fast != NULL && fast->next != NULL) {
        fast = fast->next->next;
        slow = slow->next;
    }

    // Reversing the second half of the list.
    slow = reverseList(slow, NULL);

    // Finding the maximum sum of pairs by traversing both halves of the list.
    while (slow != NULL) {
        res = res > slow->val + head->val ? res : slow->val + head->val;
        slow = slow->next;
        head = head->next;
    }

    return res;
}

// Function to create a new node with given value.
struct ListNode* createNode(int val) {
    struct ListNode* newNode = (struct ListNode*)malloc(sizeof(struct ListNode));
    newNode->val = val;
    newNode->next = NULL;
```

```c
// Function to create a new node with given value.
struct ListNode* createNode(int val) {
    struct ListNode* newNode = (struct ListNode*)malloc(sizeof(struct ListNode));
    newNode->val = val;
    newNode->next = NULL;
    return newNode;
}

// Function to add a new node at the end of the list.
void append(struct ListNode** head, int val) {
    struct ListNode* newNode = createNode(val);
    if (*head == NULL) {
        *head = newNode;
        return;
    }
    struct ListNode* temp = *head;
    while (temp->next != NULL) {
        temp = temp->next;
    }
    temp->next = newNode;
}

// Function to print the elements of the list.
void printList(struct ListNode* head) {
    while (head != NULL) {
        printf("%d ", head->val);
        head = head->next;
    }
    printf("\n");
}

```

Accepted

Sarvesh Rastogi submitted at Feb 20, 2024 12:54

Editorial Solution

Runtime

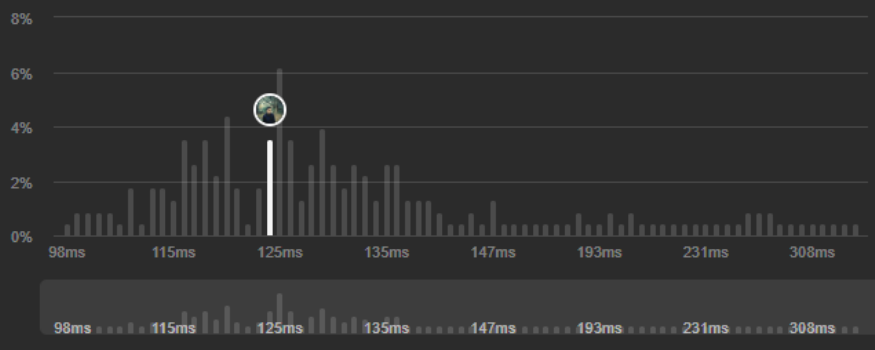
124 ms

Beats 68.28% of users with C

Memory

47.06 MB

Beats 78.41% of users with C



Code | C