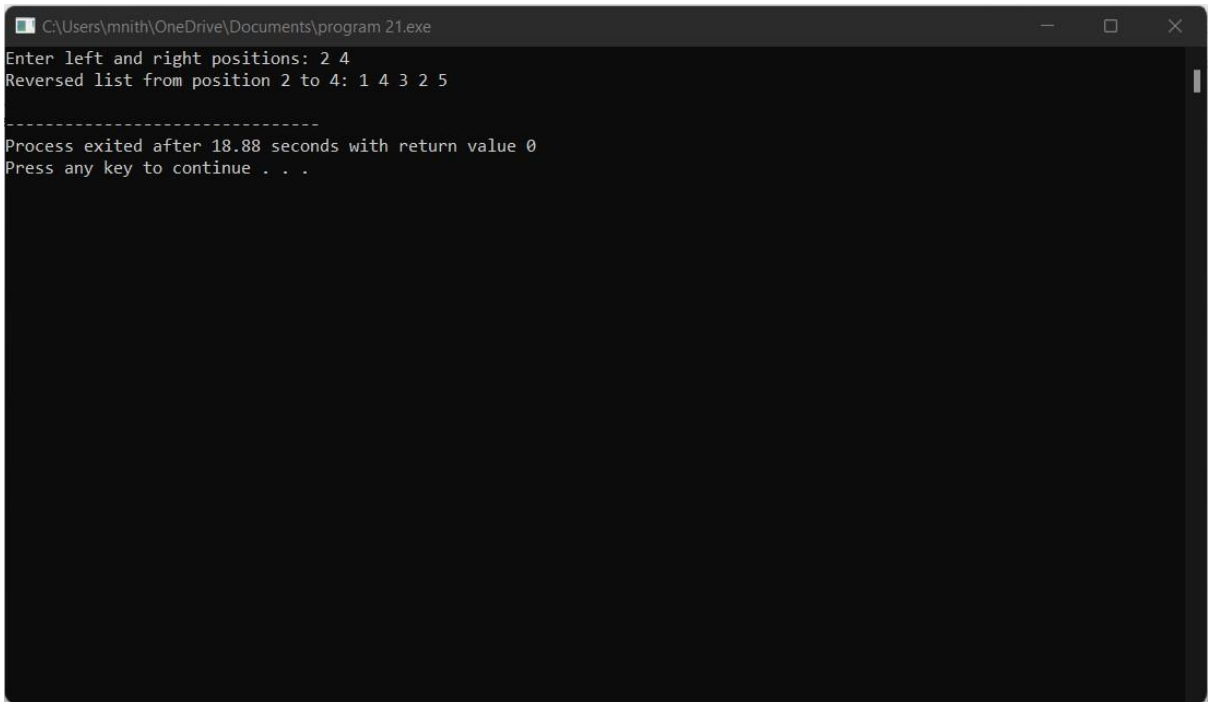
```
#include <stdio.h>
#include <stdlib.h>
struct Node {
int data;
struct Node* next;
};
int main() {
int left, right;
printf("Enter left and right positions: ");
scanf("%d %d", &left, &right);
if (left > right) {
printf("Invalid input: left should be less than or equal to right.\n");
return 1; // Exit with an error code
}
// Create a sample linked list: 1 -> 2 -> 3 -> 4 -> 5
struct Node* head = (struct Node*)malloc(sizeof(struct Node));
head->data = 1;
head->next = (struct Node*)malloc(sizeof(struct Node));
head->next->data = 2;
head->next->next = (struct Node*)malloc(sizeof(struct Node));
head->next->next->data = 3;
head->next->next->next = (struct Node*)malloc(sizeof(struct Node));
head->next->next->next->data = 4;
head->next->next->next->next = (struct Node*)malloc(sizeof(struct Node));
head->next->next->next->next->data = 5;
head->next->next->next->next->next = NULL;
struct Node* current = head;
struct Node* prev = NULL;
for (int i = 1; i < left; ++i) {
prev = current;
current = current->next;
```

```
}
struct Node* start = prev;
struct Node* end = current;
for (int i = left; i <= right; ++i) {
struct Node* nextNode = current->next;
current->next = prev;
prev = current;
current = nextNode;
}
if (start != NULL) {
start->next = prev;
} else {
head = prev;
}
end->next = current;
printf("Reversed list from position %d to %d: ", left, right);
struct Node* printNode = head;
while (printNode != NULL) {
printf("%d ", printNode->data);
printNode = printNode->next;
}
printf("\n");
struct Node* temp;
while (head != NULL) {
temp = head;
head = head->next;
free(temp);
}
return 0;
}
```


C:\Users\mnith\OneDrive\Documents\program 21.exe
Enter left and right positions: 2 4
Reversed list from position 2 to 4: 1 4 3 2 5

Process exited after 18.88 seconds with return value 0
Press any key to continue . . .