/* Program for Single linked List */ #include #include // Node structure for the linked list struct Node { int data; struct Node* next; }; // Head pointer to the list struct Node *first = NULL, *cur = NULL, *prev = NULL; // Function to add a node at the beginning of the list void addAtFirst(int value) { struct Node* newNode = (struct Node*)malloc(sizeof(struct Node)); newNode->data = value; newNode->next = first; first = newNode; printf("Node added at the beginning.\n"); } // Function to add a node at the end of the list void addAtLast(int value) { struct Node* newNode = (struct Node*)malloc(sizeof(struct Node)); newNode->data = value; newNode->next = NULL; if (first == NULL) { first = newNode; } else { cur = first; while (cur->next != NULL) { cur = cur->next; } cur->next = newNode; } printf("Node added at the end.\n"); } // Function to delete a node from the beginning of the list void deleteAtFirst() { if (first == NULL) { printf("List is empty. No nodes to delete.\n"); return; } cur = first; first = first->next; free(cur); printf("Node deleted from the beginning.\n"); } // Function to delete a node from the end of the list void deleteAtLast() { if (first == NULL) { printf("List is empty. No nodes to delete.\n"); return; } if (first->next == NULL) { free(first); first = NULL; } else { prev = first; cur = prev->next; while (cur->next != NULL) { prev = cur; cur = cur->next; } free(cur); prev->next = NULL; } printf("Node deleted from the end.\n"); } // Function to display all nodes in the list void displayList() { if (first == NULL) { printf("List is empty.\n"); return; } cur = first; printf("Linked List: "); while (cur != NULL) { printf("%d -> ", cur->data); cur = cur->next; } printf("NULL\n"); } // Main function with menu int main() { int choice, value; while (1) { printf("\nMenu:\n"); printf("1. Add at Beginning\n"); printf("2. Add at End\n"); printf("3. Delete at Beginning\n"); printf("4. Delete at End\n"); printf("5. Display List\n"); printf("6. Exit\n"); printf("Enter your choice: "); scanf("%d", &choice); switch (choice) { case 1: printf("Enter data to add at the beginning: "); scanf("%d", &value); addAtFirst(value); break; case 2: printf("\n Enter data to add at the end: "); scanf("%d", &value); addAtLast(value); break; case 3: deleteAtFirst(); break; case 4: deleteAtLast(); break; case 5: displayList(); break; case 6: exit(0); default: printf("\n Invalid choice!"); } } return 0; }