```
#include <stdio.h>
#include <stdlib.h>

struct Node {
    int data;
    struct Node* next;
};

void push(struct Node** top, int value, int* size) {
    if (*size >= 5) {
        printf("Stack is full\n");
        return;
    }
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    if (!newNode) {
        printf("Stack overflow\n");
        return;
    }
```

```
    newNode->data = value;
    newNode->next = *top;
    *top = newNode;
    (*size)++;
    printf("%d pushed to stack\n", value);
}

int pop(struct Node** top, int* size) {
    if (*top == NULL) {
        printf("Stack underflow\n");
        return -1;
    }
    struct Node* temp = *top;
    *top = (*top)->next;
    int popped = temp->data;
    free(temp);
    (*size)--;
    return popped;
}

int peek(struct Node* top) {
    if (top == NULL) {
        printf("Stack is empty\n");
        return -1;
    }
    return top->data;
}
```

```
void display(struct Node* top) {
    if (top == NULL) {
        printf("Stack is empty\n");
        return;
    }
    printf("Stack elements: ");
    struct Node* temp = top;
    while (temp != NULL) {
        printf("%d ", temp->data);
        temp = temp->next;
    }
    printf("\n");
}

int main() {
    struct Node* top = NULL;
    int choice, value;
    int size = 0;
    printf("Stack Operations using Linked List (Size limit: 5):\n");
    printf("1. Push\n2. Pop\n3. Peek\n4. Display\n5. Exit\n");
    while (1) {
        printf("\nEnter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter value to push: ");
                scanf("%d", &value);
```

```
            push(&top, value, &size);
            break;
        case 2:
            value = pop(&top, &size);
            if (value != -1) {
                printf("Popped value: %d\n", value);
            }
            break;
        case 3:
            value = peek(top);
            if (value != -1) {
                printf("Top element: %d\n", value);
            }
            break;
        case 4:
            display(top);
            break;
        case 5:
            printf("Exiting...\n");
            return 0;
        default:
            printf("Invalid choice! Please try again.\n");
      }
   }

   return 0;
}
```