WAP to Implement Single Link List with following operations: Sort the linked list, Reverse the linked list, Concatenation of two linked lists.

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

/* Node structure */
struct node {
   int data;
   struct node *next;
};

/* Function declarations */
struct node* createNode(int data);
void insertEnd(struct node **head, int data);
void display(struct node *head);
void sortList(struct node *head);
void reverseList(struct node **head);
void concatenate(struct node **head1, struct node *head2);

/* Create new node */
struct node* createNode(int data) {
   struct node *newNode = (struct node *)malloc(sizeof(struct node));
   newNode->data = data;
   newNode->next = NULL;
   return newNode;
}

/* Insert at end */
void insertEnd(struct node **head, int data) {
   struct node *newNode = createNode(data);
```

```
    struct node *temp = *head;

    if (*head == NULL) {
        *head = newNode;
        return;
    }

    while (temp->next != NULL)
        temp = temp->next;

    temp->next = newNode;
}

/* Display list */
void display(struct node *head) {
    if (head == NULL) {
        printf("List is empty\n");
        return;
    }
    while (head != NULL) {
        printf("%d -> ", head->data);
        head = head->next;
    }
    printf("NULL\n");
}

/* Sort list */
void sortList(struct node *head) {
    struct node *i, *j;
    int temp;
```

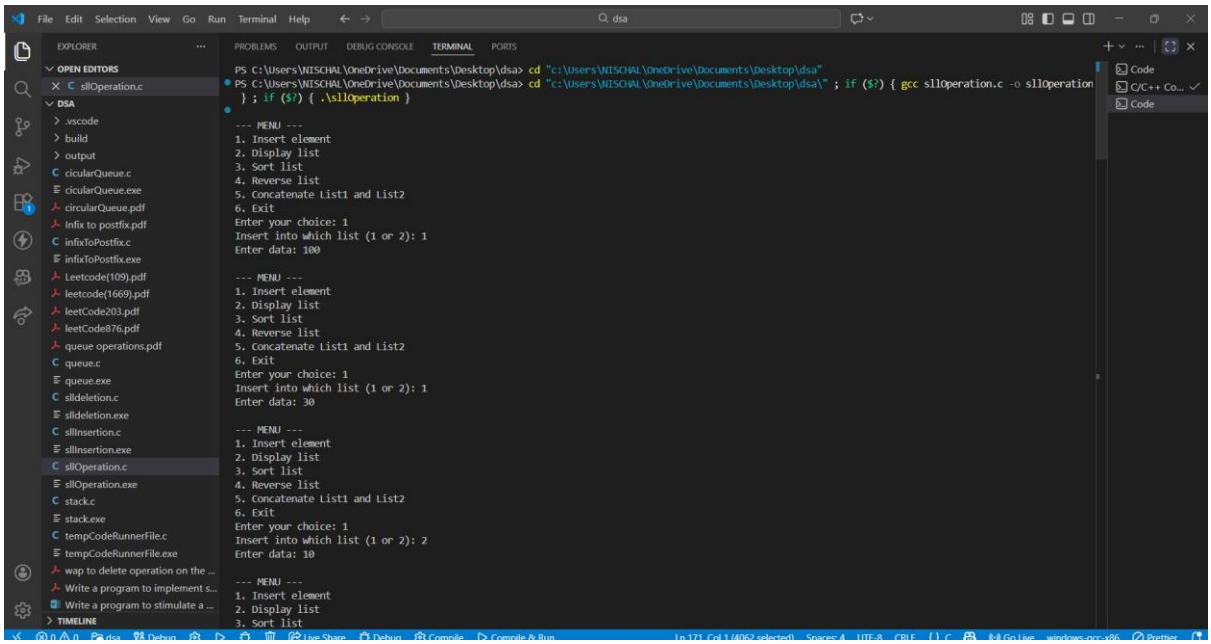
```
    if (head == NULL)
        return;

    for (i = head; i->next != NULL; i = i->next) {
        for (j = i->next; j != NULL; j = j->next) {
            if (i->data > j->data) {
                temp = i->data;
                i->data = j->data;
                j->data = temp;
            }
        }
    }
    printf("List sorted successfully.\n");
}

/* Reverse list */
void reverseList(struct node **head) {
    struct node *prev = NULL, *current = *head, *next = NULL;

    while (current != NULL) {
        next = current->next;
        current->next = prev;
        prev = current;
        current = next;
    }
    *head = prev;
    printf("List reversed successfully.\n");
}

/* Concatenate two lists */
void concatenate(struct node **head1, struct node *head2) {
```

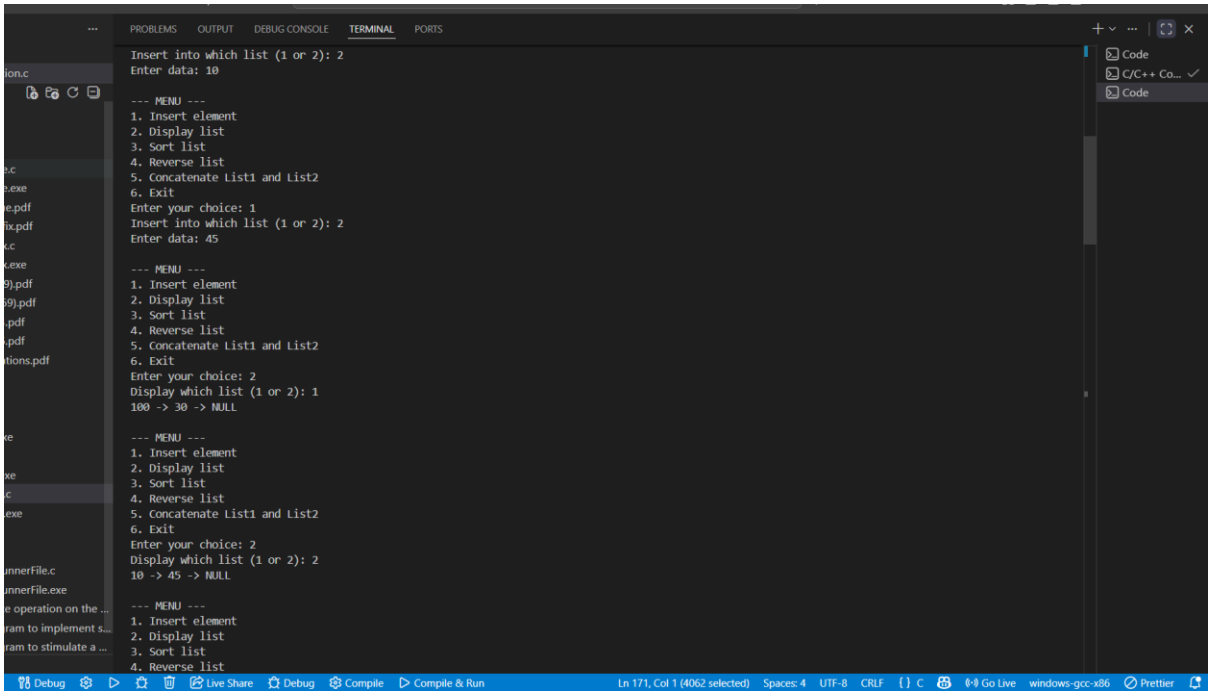
```
    struct node *temp = *head1;

    if (*head1 == NULL) {
        *head1 = head2;
        return;
    }

    while (temp->next != NULL)
        temp = temp->next;

    temp->next = head2;
    printf("Lists concatenated successfully.\n");
}

/* Main function */
int main() {
    struct node *list1 = NULL, *list2 = NULL;
    int choice, data, listChoice;

    do {
        printf("\n--- MENU ---\n");
        printf("1. Insert element\n");
        printf("2. Display list\n");
        printf("3. Sort list\n");
        printf("4. Reverse list\n");
        printf("5. Concatenate List1 and List2\n");
        printf("6. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
```

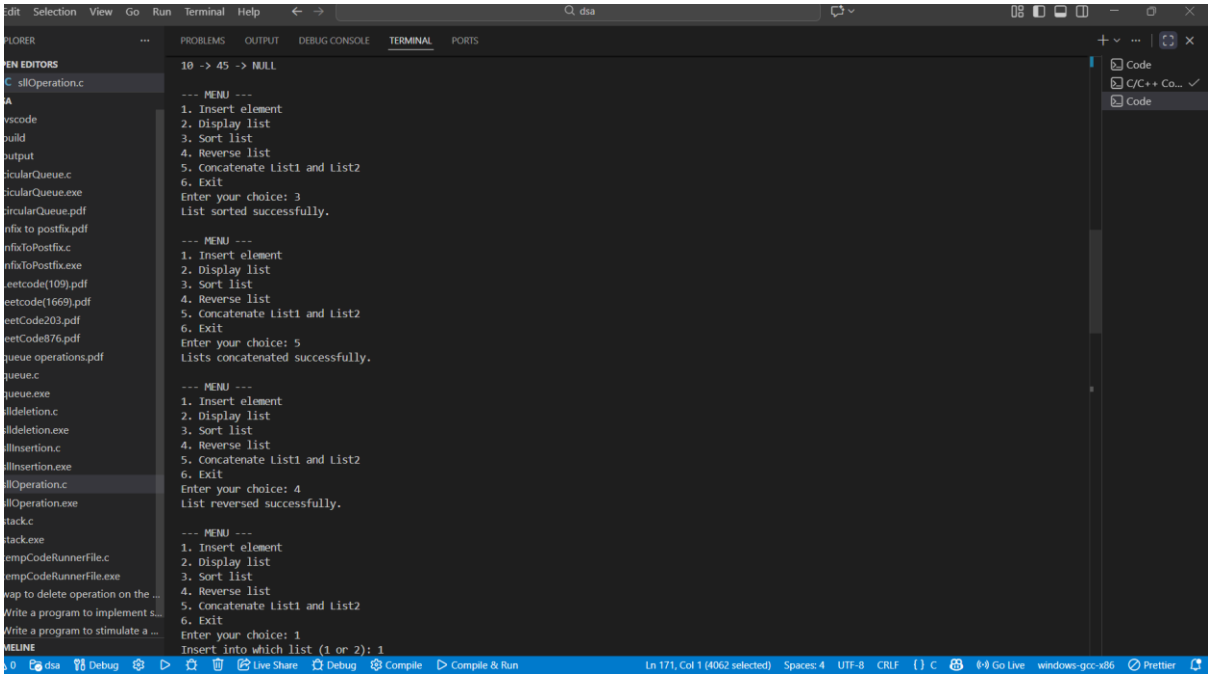
```
case 1:
   printf("Insert into which list (1 or 2): ");
   scanf("%d", &listChoice);
   printf("Enter data: ");
   scanf("%d", &data);

   if (listChoice == 1)
      insertEnd(&list1, data);
   else if (listChoice == 2)
      insertEnd(&list2, data);
   else
      printf("Invalid list choice\n");
   break;

case 2:
   printf("Display which list (1 or 2): ");
   scanf("%d", &listChoice);

   if (listChoice == 1)
      display(list1);
   else if (listChoice == 2)
      display(list2);
   else
      printf("Invalid list choice\n");
   break;

case 3:
   sortList(list1);
   break;

case 4:
```

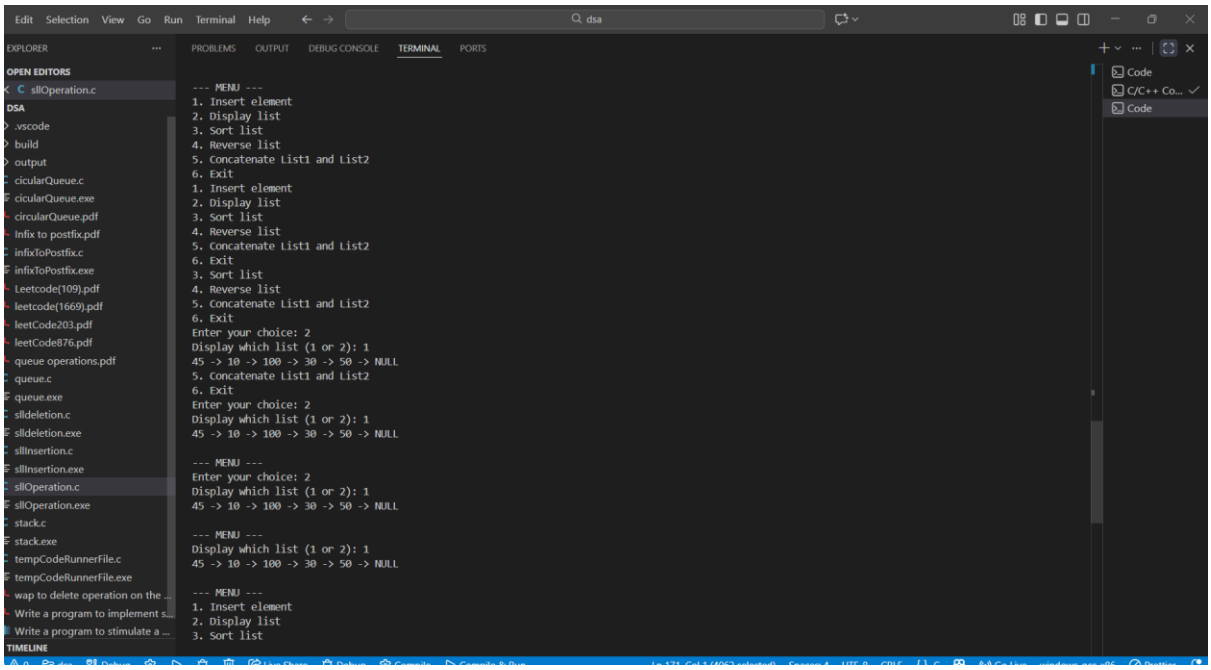
```
            reverseList(&list1);
            break;

        case 5:
            concatenate(&list1, list2);
            break;

        case 6:
            printf("Exiting program.\n");
            break;

        default:
            printf("Invalid choice!\n");
        }
    } while (choice != 6);

    return 0;
}
```

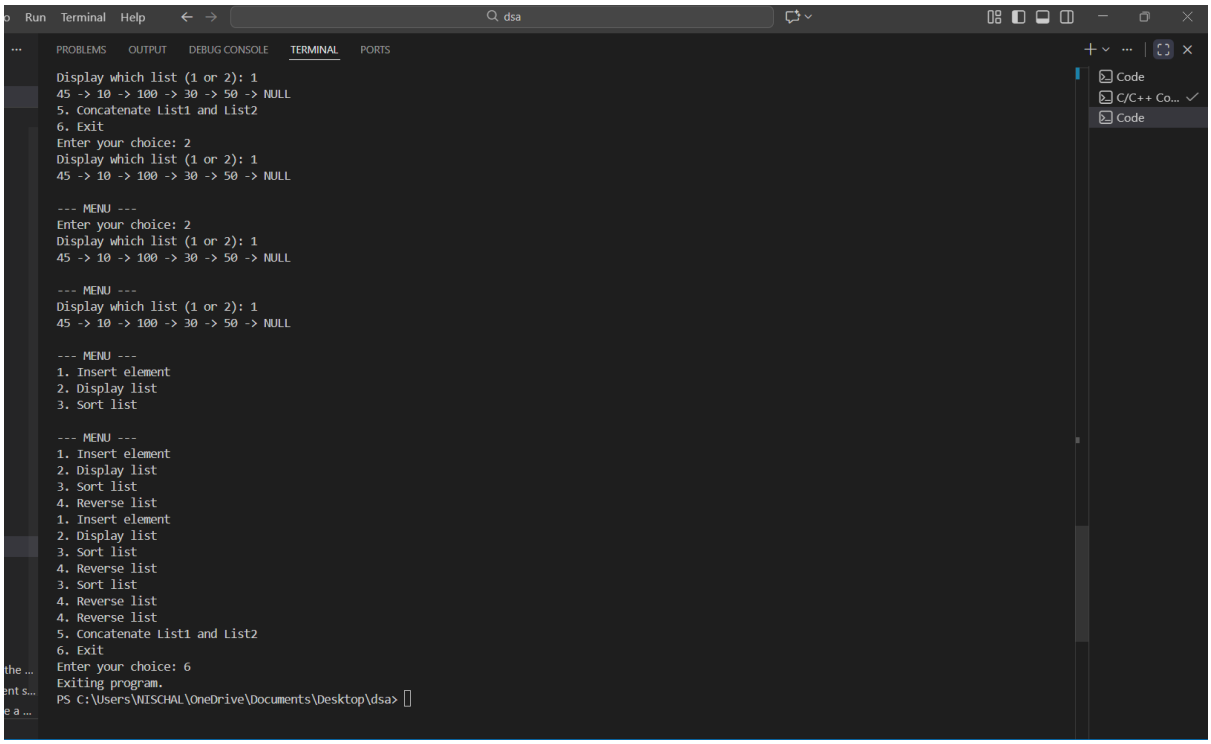
OUTPUT:




PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Insert into which list (1 or 2): 2
Enter data: 10
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 1
Insert into which list (1 or 2): 2
Enter data: 45
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 2
Display which list (1 or 2): 1
100 -> 30 -> NULL
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 2
Display which list (1 or 2): 2
10 -> 45 -> NULL
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
Code
C/C++ Co...
Code
Debug
Live Share
Debug
Compile
Compile & Run
Ln 171, Col 1 (4062 selected)
Spaces: 4
UTF-8
CRLF
C
Go Live
windows-gcc-x86
Prettier


Edit
Selection
View
Go
Run
Terminal
Help
dsa
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
sllOperation.c
10 -> 45 -> NULL
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 3
List sorted successfully.
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 5
Lists concatenated successfully.
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 4
List reversed successfully.
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 1
Insert into which list (1 or 2): 1
Code
C/C++ Co...
Code
dsa
Debug
Live Share
Debug
Compile
Compile & Run
Ln 171, Col 1 (4062 selected)
Spaces: 4
UTF-8
CRLF
C
Go Live
windows-gcc-x86
Prettier


Edit Selection View Go Run Terminal Help
dsa
EXPLORER
OPEN EDITORS
sllOperation.c
DSA
.vscode
build
output
cicularQueue.c
cicularQueue.exe
circularQueue.pdf
Infix to postfix.pdf
infixToPostfix.c
infixToPostfix.exe
Leetcode(109).pdf
leetcode(1669).pdf
leetCode203.pdf
leetCode876.pdf
queue operations.pdf
queue.c
queue.exe
slldeletion.c
slldeletion.exe
sllInsertion.c
sllInsertion.exe
sllOperation.c
sllOperation.exe
stack.c
stack.exe
tempCodeRunnerFile.c
tempCodeRunnerFile.exe
wap to delete operation on the ...
Write a program to implement s...
Write a program to stimulate a ...
TIMELINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
1. Insert element
2. Display list
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
3. Sort list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 2
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
5. Concatenate List1 and List2
6. Exit
Enter your choice: 2
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
--- MENU ---
Enter your choice: 2
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
--- MENU ---
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
--- MENU ---
1. Insert element
2. Display list
3. Sort list
Code
C/C++ Co...
Code


Run Terminal Help
dsa
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
5. Concatenate List1 and List2
6. Exit
Enter your choice: 2
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
--- MENU ---
Enter your choice: 2
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
--- MENU ---
Display which list (1 or 2): 1
45 -> 10 -> 100 -> 30 -> 50 -> NULL
--- MENU ---
1. Insert element
2. Display list
3. Sort list
--- MENU ---
1. Insert element
2. Display list
3. Sort list
4. Reverse list
1. Insert element
2. Display list
3. Sort list
4. Reverse list
3. Sort list
4. Reverse list
4. Reverse list
5. Concatenate List1 and List2
6. Exit
Enter your choice: 6
Exiting program.
PS C:\Users\NISCHAL\OneDrive\Documents\Desktop\dsa>
Code
C/C++ Co...
Code