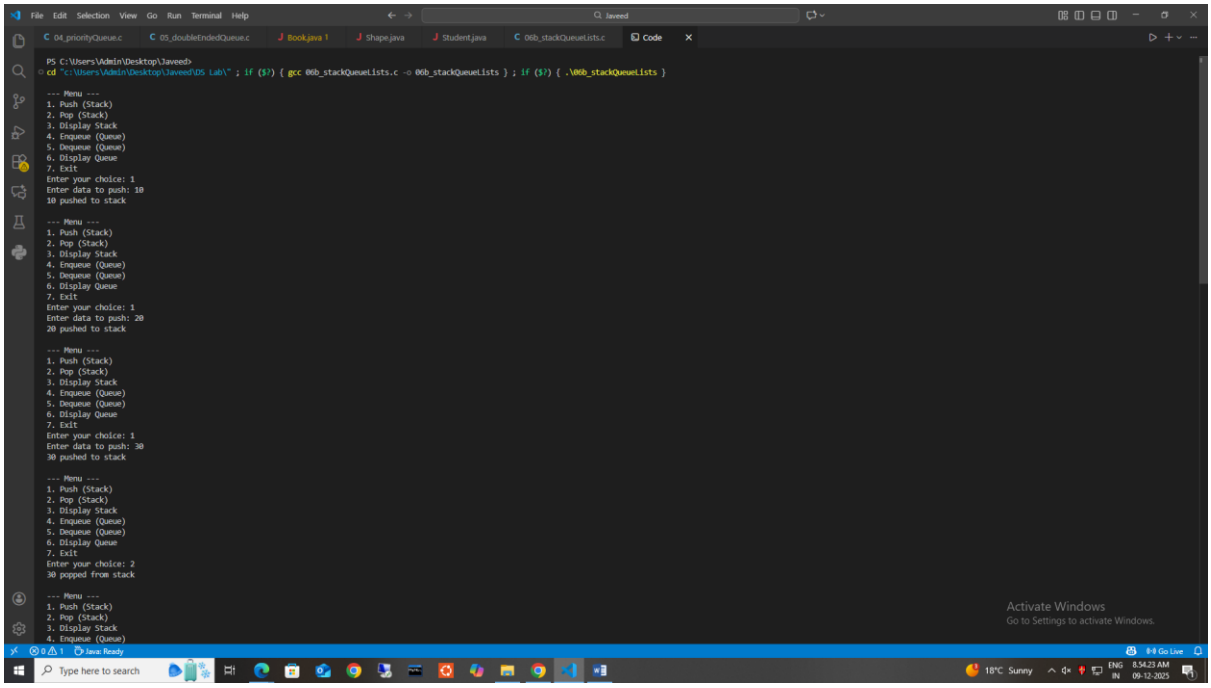
6b. WAP to Implement Single Link List to simulate Stack & Queue Operations.

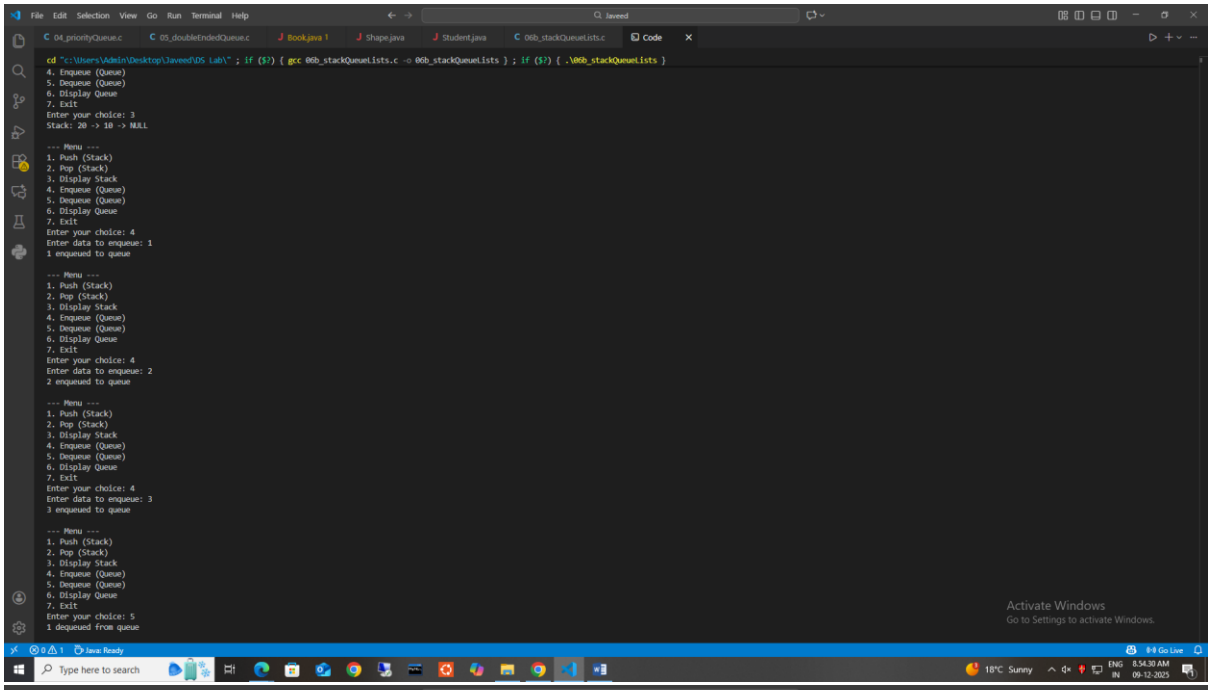
```c
#include <stdio.h>
#include <stdlib.h>

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

// Function to create a new node
struct Node* createNode(int data) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = data;
    newNode->next = NULL;
    return newNode;
}

// ---------- STACK OPERATIONS ----------
void push(struct Node** top, int data) {
    struct Node* newNode = createNode(data);
    newNode->next = *top;
    *top = newNode;
    printf("%d pushed to stack\n", data);
}

void pop(struct Node** top) {
    if (*top == NULL) {
        printf("Stack Underflow\n");
        return;
    }
    struct Node* temp = *top;
    *top = (*top)->next;
    printf("%d popped from stack\n", temp->data);
    free(temp);
}

// ---------- QUEUE OPERATIONS ----------
void enqueue(struct Node** front, struct Node** rear, int data) {
    struct Node* newNode = createNode(data);
    if (*rear == NULL) {
        *front = *rear = newNode;
    } else {
        (*rear)->next = newNode;
        *rear = newNode;
    }
```

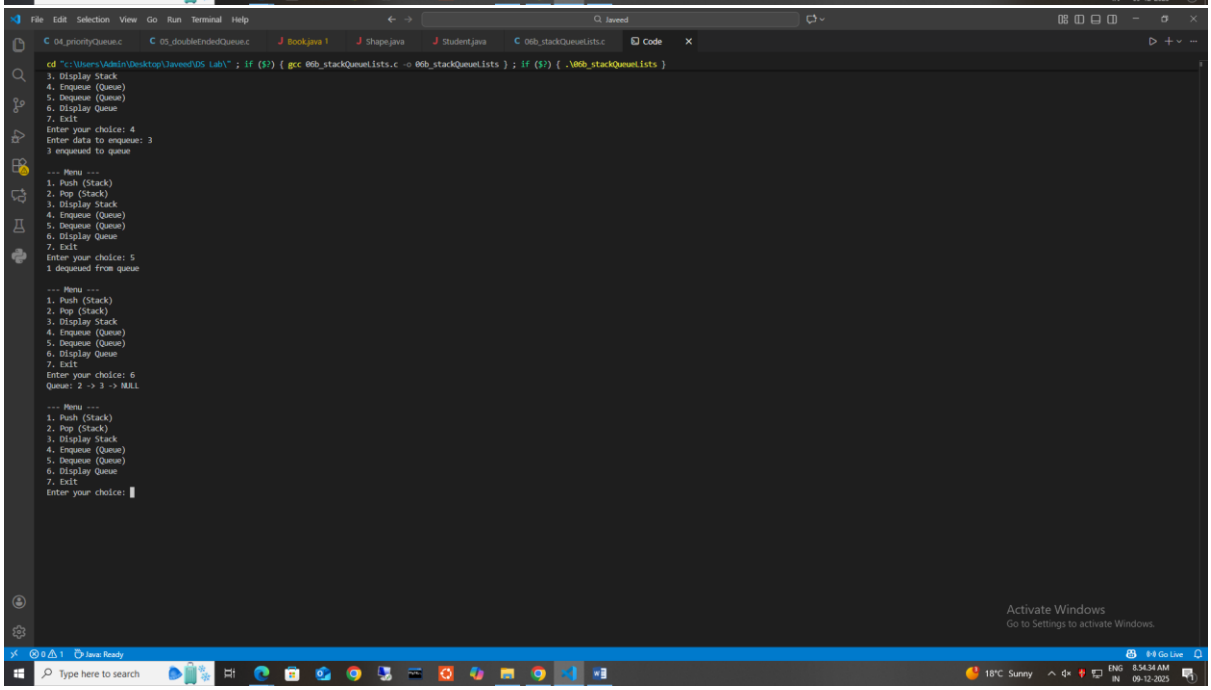
```c
    printf("%d enqueued to queue\n", data);
}

void dequeue(struct Node** front, struct Node** rear) {
    if (*front == NULL) {
        printf("Queue Underflow\n");
        return;
    }
    struct Node* temp = *front;
    *front = (*front)->next;
    if (*front == NULL) {
        *rear = NULL;
    }
    printf("%d dequeued from queue\n", temp->data);
    free(temp);
}

// ---------- DISPLAY ----------
void display(struct Node* head) {
    if (head == NULL) {
        printf("Empty\n");
        return;
    }
    struct Node* temp = head;
    while (temp != NULL) {
        printf("%d -> ", temp->data);
        temp = temp->next;
    }
    printf("NULL\n");
}

// ---------- MAIN ----------
int main() {
    struct Node* stackTop = NULL;       // Stack top
    struct Node* queueFront = NULL;     // Queue front
    struct Node* queueRear = NULL;      // Queue rear

    int choice, data;

    while (1) {
        printf("\n--- Menu ---\n");
        printf("1. Push (Stack)\n");
        printf("2. Pop (Stack)\n");
        printf("3. Display Stack\n");
        printf("4. Enqueue (Queue)\n");
        printf("5. Dequeue (Queue)\n");
        printf("6. Display Queue\n");
        printf("7. Exit\n");
```

```
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter data to push: ");
                scanf("%d", &data);
                push(&stackTop, data);
                break;
            case 2:
                pop(&stackTop);
                break;
            case 3:
                printf("Stack: ");
                display(stackTop);
                break;
            case 4:
                printf("Enter data to enqueue: ");
                scanf("%d", &data);
                enqueue(&queueFront, &queueRear, data);
                break;
            case 5:
                dequeue(&queueFront, &queueRear);
                break;
            case 6:
                printf("Queue: ");
                display(queueFront);
                break;
            case 7:
                printf("Exiting...\n");
                exit(0);
            default:
                printf("Invalid choice! Try again.\n");
        }
    }

    return 0;
}
```

OUTPUT:


PS C:\Users\Admin\Desktop\Javeed>
cd "c:\Users\Admin\Desktop\Javeed\DS Lab\" ; if ($?) { gcc 06b_stackQueueLists.c -o 06b_stackQueueLists } ; if ($?) { .\06b_stackQueueLists }
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 1
Enter data to push: 10
10 pushed to stack
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 1
Enter data to push: 20
20 pushed to stack
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 1
Enter data to push: 30
30 pushed to stack
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 2
30 popped from stack
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
Activate Windows
Go to Settings to activate Windows.


cd "c:\Users\Admin\Desktop\Javeed\DS Lab\" ; if ($?) { gcc 06b_stackQueueLists.c -o 06b_stackQueueLists } ; if ($?) { .\06b_stackQueueLists }
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 3
Stack: 20 -> 10 -> NULL
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 4
Enter data to enqueue: 1
1 enqueued to queue
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 4
Enter data to enqueue: 2
2 enqueued to queue
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 4
Enter data to enqueue: 3
3 enqueued to queue
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 5
1 dequeued from queue
Activate Windows
Go to Settings to activate Windows.


cd "c:\Users\Admin\Desktop\Javeed\DS Lab\" ; if ($?) { gcc 06b_stackQueueLists.c -o 06b_stackQueueLists } ; if ($?) { .\06b_stackQueueLists }
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 4
Enter data to enqueue: 3
3 enqueued to queue
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 5
1 dequeued from queue
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice: 6
Queue: 2 -> 3 -> NULL
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter your choice:
Activate Windows
Go to Settings to activate Windows.