Name: Vivek Vishnu Lele

Roll No: 23MCA1023

Topic: Queue using link list

Code**:**

**#include <stdio.h>**

**#include <stdlib.h>**

**struct Node {**

**int data;**

**struct Node\* next;**

**};**

**struct Queue {**

**struct Node\* front;**

**struct Node\* rear;**

**};**

**struct Node\* createNode(int data) {**

**struct Node\* newNode = (struct Node\*)malloc(sizeof(struct Node));**

**newNode->data = data;**

**newNode->next = NULL;**

**return newNode;**

**}**

**struct Queue\* createQueue() {**

**struct Queue\* queue = (struct Queue\*)malloc(sizeof(struct Queue));**

**queue->front = queue->rear = NULL;**

**return queue;**

**}**

**int isEmpty(struct Queue\* queue) {**

**return queue->front == NULL;**

**}**

**void enqueue(struct Queue\* queue, int data) {**

**struct Node\* newNode = createNode(data);**

**if (queue->rear == NULL) {**

**queue->front = queue->rear = newNode;**

**return;**

**}**

**queue->rear->next = newNode;**

**queue->rear = newNode;**

**}**

**int dequeue(struct Queue\* queue) {**

**if (isEmpty(queue)) {**

**return -1; // Queue is empty**

**}**

**struct Node\* temp = queue->front;**

**int data = temp->data;**

**queue->front = queue->front->next;**

**if (queue->front == NULL) {**

**queue->rear = NULL;**

**}**

**free(temp);**

**return data;**

**}**

**int peek(struct Queue\* queue) {**

**if (isEmpty(queue)) {**

**return -1; // Queue is empty**

**}**

**return queue->front->data;**

**}**

**void display(struct Queue\* queue) {**

**struct Node\* current = queue->front;**

**while (current != NULL) {**

**printf("%d ", current->data);**

**current = current->next;**

**}**

**printf("\n");**

**}**

**int main() {**

**struct Queue\* queue = createQueue();**

**while (1) {**

**printf("\nQueue Operations:\n");**

**printf("1. Enqueue\n");**

**printf("2. Dequeue\n");**

**printf("3. Peek\n");**

**printf("4. Display\n");**

**printf("5. Exit\n");**

**int choice;**

**printf("Enter your choice: ");**

**scanf("%d", &choice);**

**switch (choice) {**

**case 1: {**

**int data;**

**printf("Enter the value to enqueue: ");**

**scanf("%d", &data);**

**enqueue(queue, data);**

**break;**

**}**

**case 2: {**

**int dequeuedData = dequeue(queue);**

**if (dequeuedData != -1) {**

**printf("Dequeued: %d\n", dequeuedData);**

**} else {**

**printf("Queue is empty.\n");**

**}**

**break;**

**}**

**case 3: {**

**int peekData = peek(queue);**

**if (peekData != -1) {**

**printf("Peek: %d\n", peekData);**

**} else {**

**printf("Queue is empty.\n");**

**}**

**break;**

**}**

**case 4:**

**display(queue);**

**break;**

**case 5:**

**printf("Exiting...\n");**

**exit(0);**

**default:**

**printf("Invalid choice. Please enter a valid option.\n");**

**}**

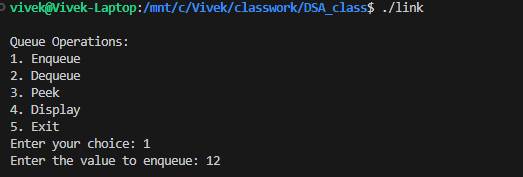
**}**

**return 0;**

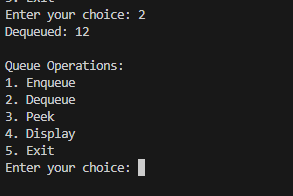
**}**

**Output:**

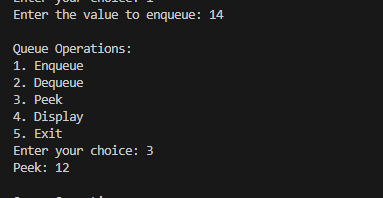
**1. Enqueue**

****

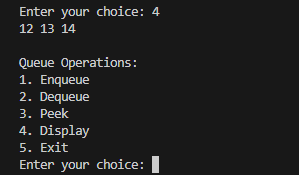
**Dequeue:**

****

**Peek:**

****

**Display:**

****