#include<stdio.h>

#include <stdlib.h>

struct Node {

int data;

struct Node\* next;

};

struct Node\* front = NULL;

struct Node\* rear = NULL;

void enqueue(int element) {

struct Node\* new\_node = (struct Node\*)malloc(sizeof(struct Node));

new\_node->data = element;

new\_node->next = NULL;

if (front == NULL && rear == NULL) {

front = rear = new\_node;

return;

}

rear->next = new\_node;

rear = new\_node;

}

int dequeue() {

if (front == NULL) {

printf("Queue is empty");

return -1;

}

struct Node\* temp = front;

int element = temp->data;

if (front == rear) {

front = rear = NULL;

}

else {

front = front->next;

}

free(temp);

return element;

}

void display() {

struct Node\* temp;

if ((front == NULL) && (rear == NULL)) {

printf("\nQueue is Empty\n");

} else {

printf("The queue is \n");

temp = front;

while (temp) {

printf("%d--->", temp -> data);

temp = temp -> next;

}

printf("NULL\n");

}

}

int main() {

int choice, value;

while(1) {

printf("Queue operations:\n");

printf("1.Enqueue\n");

printf("2.Dequeue\n");

printf("3.Display\n");

printf("4.Exit\n");

printf("Enter choice: ");

scanf("%d",&choice);

switch(choice)

{

case 1:

printf("Enter value:");

scanf("%d",&value);

enqueue(value);

break;

case 2:

dequeue();

display();

break;

case 3:

printf("Your queue:");

display();

break;

case 4:

printf("Exiting program...");

return 0;

default:

printf("\nInvalid choice!\n");

}

}

return 0;

}