**DS project**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define SIZE 5 // Maximum queue size

struct Queue {

char names[SIZE][50];

int front;

int rear;

};

void initialize(struct Queue \*q) {

q->front = -1;

q->rear = -1;

}

int isFull(struct Queue \*q) {

return q->rear == SIZE - 1;

}

int isEmpty(struct Queue \*q) {

return q->front == -1 || q->front > q->rear;

}

void enqueue(struct Queue \*q, char name[]) {

if (isFull(q)) {

printf("\nQueue is full! No more tickets can be booked.\n");

return;

}

if (q->front == -1)

q->front = 0;

q->rear++;

strcpy(q->names[q->rear], name);

printf("\nTicket booked successfully for %s.\n", name);

}

void dequeue(struct Queue \*q) {

if (isEmpty(q)) {

printf("\nNo customers in queue!\n");

return;

}

printf("\nTicket served for %s.\n", q->names[q->front]);

q->front++;

}

void display(struct Queue \*q) {

if (isEmpty(q)) {

printf("\nNo customers waiting in queue.\n");

return;

}

printf("\nCurrent queue:\n");

for (int i = q->front; i <= q->rear; i++) {

printf("%d. %s\n", i - q->front + 1, q->names[i]);

}

}

int main() {

struct Queue q;

initialize(&q);

int choice;

char name[50];

printf("=== Ticket Booking System Using Queue ===\n");

while (1) {

printf("\n1. Book Ticket (Enqueue)");

printf("\n2. Serve Customer (Dequeue)");

printf("\n3. Display Queue");

printf("\n4. Exit");

printf("\nEnter your choice: ");

scanf("%d", &choice);

getchar(); // Clear newline from input buffer

switch (choice) {

case 1:

printf("Enter customer name: ");

gets(name);

enqueue(&q, name);

break;

case 2:

dequeue(&q);

break;

case 3:

display(&q);

break;

case 4:

printf("\nExiting... Thank you!\n");

exit(0);

default:

printf("\nInvalid choice! Try again.\n");

}

}

return 0;

}