***Question 1:***

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

typedef struct Participant {

char name[30];

int age;

int registrationNumber;

struct Participant\* next;

} Participant;

typedef struct {

Participant\* front;

Participant\* rear;

} Queue;

void initializeQueue(Queue\* q) {

q->front = q->rear = NULL;

}

Participant\* createParticipant(char\* name, int age, int regNum) {

Participant\* newParticipant = (Participant\*)malloc(sizeof(Participant));

strcpy(newParticipant->name, name);

newParticipant->age = age;

newParticipant->registrationNumber = regNum;

newParticipant->next = NULL;

return newParticipant;

}

void enqueue(Queue\* q, char\* name, int age, int regNum) {

Participant\* newParticipant = createParticipant(name, age, regNum);

if (q->rear == NULL) {

q->front = q->rear = newParticipant;

return;

}

q->rear->next = newParticipant;

q->rear = newParticipant;

}

void dequeue(Queue\* q) {

if (q->front == NULL) {

printf("The queue is empty. No participants to process.\n");

return;

}

Participant\* temp = q->front;

q->front = q->front->next;

if (q->front == NULL) {

q->rear = NULL;

}

printf("Processed participant: %s, Age: %d, Registration Number: %d\n", temp->name, temp->age, temp->registrationNumber);

free(temp);

}

void peek(Queue\* q) {

if (q->front == NULL) {

printf("The queue is empty.\n");

} else {

printf("Next participant in line: %s, Age: %d, Registration Number: %d\n", q->front->name, q->front->age, q->front->registrationNumber);

}

}

void displayQueue(Queue\* q) {

if (q->front == NULL) {

printf("The queue is empty.\n");

return;

}

Participant\* temp = q->front;

printf("Participants in the queue:\n");

while (temp != NULL) {

printf("Name: %s, Age: %d, Registration Number: %d\n", temp->name, temp->age, temp->registrationNumber);

temp = temp->next;

}

}

void freeQueue(Queue\* q) {

Participant\* temp = q->front;

while (temp != NULL) {

Participant\* next = temp->next;

free(temp);

temp = next;

}

q->front = q->rear = NULL;

}

int main() {

Queue q;

initializeQueue(&q);

int choice;

char name[50];

int age, regNum;

printf("Event Management System\n");

printf("Anushka Rai\n");

printf("USN: 1AY23CS034\n\n");

do {

printf("Event Management System Menu:\n");

printf("1. Register participant (Enqueue)\n");

printf("2. Process participant (Dequeue)\n");

printf("3. View next participant (Peek)\n");

printf("4. Display all participants\n");

printf("5. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("Enter participant name: ");

scanf("%s", name);

printf("Enter participant age: ");

scanf("%d", &age);

printf("Enter registration number: ");

scanf("%d", &regNum);

enqueue(&q, name, age, regNum);

break;

case 2:

dequeue(&q);

break;

case 3:

peek(&q);

break;

case 4:

displayQueue(&q);

break;

case 5:

freeQueue(&q);

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

break;

default:

printf("Invalid choice. Please try again.\n");

}

} while (choice != 5);

return 0;

}

***OUTPUT:***

Event Management System

Anushka Rai

USN: 1AY23CS034

Event Management System Menu:

1. Register participant (Enqueue)

2. Process participant (Dequeue)

3. View next participant (Peek)

4. Display all participants

5. Exit

Enter your choice: 1

Enter participant name: Anushka

Enter participant age: 19

Enter registration number: 034

Event Management System Menu:

1. Register participant (Enqueue)

2. Process participant (Dequeue)

3. View next participant (Peek)

4. Display all participants

5. Exit

Enter your choice: 1

Enter participant name: Anusha

Enter participant age: 19

Enter registration number: 033

Event Management System Menu:

1. Register participant (Enqueue)

2. Process participant (Dequeue)

3. View next participant (Peek)

4. Display all participants

5. Exit

Enter your choice: 3

Next participant in line: Anushka, Age: 19, Registration Number: 34

Event Management System Menu:

1. Register participant (Enqueue)

2. Process participant (Dequeue)

3. View next participant (Peek)

4. Display all participants

5. Exit

Enter your choice: 4

Participants in the queue:

Name: Anushka, Age: 19, Registration Number: 34

Name: Anusha, Age: 19, Registration Number: 33

Event Management System Menu:

1. Register participant (Enqueue)

2. Process participant (Dequeue)

3. View next participant (Peek)

4. Display all participants

5. Exit

Enter your choice: 5

Exiting...  
  
  
***Question 2:***   
  
#include <stdio.h>

#include <string.h>

void readInput(char\* sentence, char\* word);

void deleteWord(char\* sentence, const char\* word);

void displayResult(const char\* sentence);

int main() {

char sentence[1000];

char word[100];

printf("Word Deletion from a Sentence\n");

printf("Anushka Rai\n");

printf("USN: 1AY23CS034\n\n");

readInput(sentence, word);

deleteWord(sentence, word);

displayResult(sentence);

return 0;

}

void readInput(char\* sentence, char\* word) {

printf("Enter a sentence: ");

fgets(sentence, 1000, stdin);

sentence[strcspn(sentence, "\n")] = '\0';

printf("Enter the word to delete: ");

scanf("%s", word);

}

void deleteWord(char\* sentence, const char\* word) {

char\* pos;

int len = strlen(word);

while ((pos = strstr(sentence, word)) != NULL) {

memmove(pos, pos + len, strlen(pos + len) + 1);

}

}

void displayResult(const char\* sentence) {

if (strlen(sentence) > 0) {

printf("Modified sentence: %s\n", sentence);

} else {

printf("No deletions were made. The sentence does not contain the specified word.\n");

}

}

***OUTPUT:***Word Deletion from a Sentence

Anushka Rai

USN: 1AY23CS034

Enter a sentence: Today is a nice day!

Enter the word to delete: nice

Modified sentence: Today is a day!