PROJECT REPORT

DATA STRUCTURES AND ALGORITHMS (UCS049)

TOPIC: IMPLEMENTATION OF RAILWAY TICKET RESERVATION SYSTEM

SUBMITTED TO: DR. DIWAKAR TRIPATHI

SUBMITTED BY: ANKIT KUMAR (101804042)
SAMRIDH SHARMA (101804046)
ARSH GUPTA (101804066)
SAHARSH HANDA (101804067)

ELECTRICAL AND INSTRUMENTATION ENGINEERING DEPARTMENT

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

PATIALA – 147004, PUNJAB

MARCH-JULY 2021

ABSTRACT

This project take the basic details of a passenger as input such as name, age, gender and it's berth preference for reservation of seat in a train system.

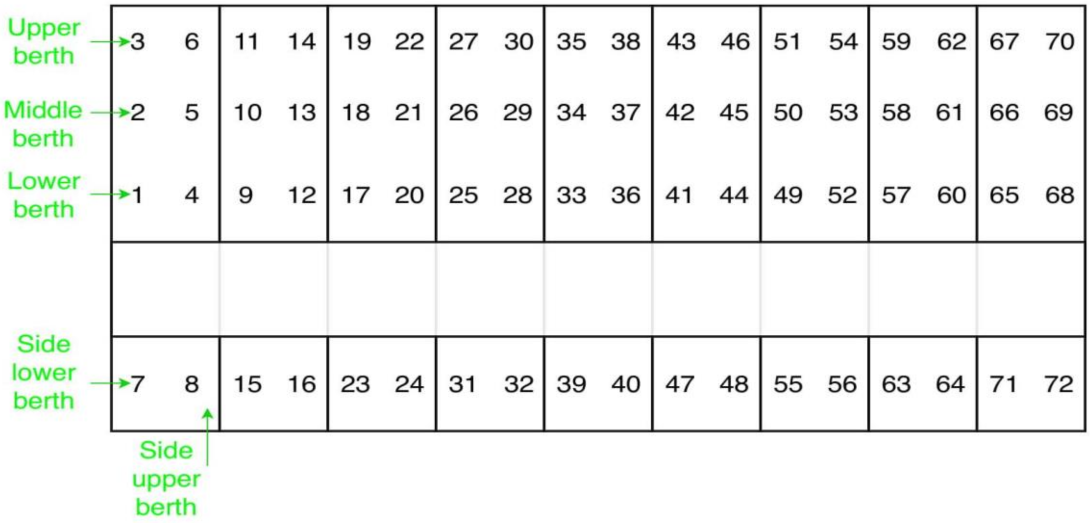
This System is implemented using classes, arrays and switch case instead of linkedlist. Linked list is not used because of its complexity. For example if seat number is provided by the passenger, it points straight towards the class objects related to same, but in case of linkedlist, whole is to be traversed. Even during cancellation provided seat number is searched throughout and then it can be deleted.

For successful completion of this mini project, we also took few assumptions which are discussed in next page. Through dynamic memory allocation, class pointers and switch case statements, we can easily move around the objects.

ASSUMPTIONS

Different assumptions were taken for this project which are as follows:

1. We considered only a single coach for this implementation containing 72 seats only. Its architecture is shown in the figure below.



3	6	11	14	19	22	27	30	35	38	43	46	51	54	59	62	67	70
2	5	10	13	18	21	26	29	34	37	42	45	50	53	58	61	66	69
1	4	9	12	17	20	25	28	33	36	41	44	49	52	57	60	65	68
7	8	15	16	23	24	31	32	39	40	47	48	55	56	63	64	71	72

2. This program do not follow database management system.

3. The payment process for a seat reservation is considered assuccessful with the booking of a particular seat by the user.

4. After cancellation of seat, first it'll book the seats after the current booked seat, and then when all the seats are filled for the berth, it'll check for any vacant available seat and will then fill it.

IMPLEMENTATION

- The system is based on a concept to reserve train tickets. Here, the program is executed, there's a system where the user is asked to select the one choice among reservation of a seat, cancellation of a seat and the booked ticket details.
- For reservation of a seat, the function **void reserve()** is used which ask the user to enter his/her details like name, age, gender etc.. as according to the **class alldetails**.
- Then the user is asked to choose the berth preference and for this the functions like **void booklower(obj), void bookmiddle(obj), void bookupper(obj), void booksupper(obj),void bookslower(obj), void bookrandom(obj)** are used and according to the user's choice the seat number in that berth is allotted.
- For cancellation of a seat, the function **void cancel()** is used which ask the user to enter his/her reserved seat number and then that seat is cancelled.
- For showcase of booked ticket details, the function **void details()** is used which ask the user to enter his/her reserved seat number then it displays the corresponding details to that seat number.
- This shows the systematic presentation of the complete ticket reservation system.

Code

```
#include <bits/stdc++.h>

using namespace std;

class alldetails{

        public:

                int seatnumber;

                int age;

                char gender;

                string name_ls;

                string berth_info;

};

alldetails* arr[73]={NULL};

int a = 0;

int b = 0,c = 0,d = 0,e = 0;

int lower[20];

        int middle[20];

        int upper[20];
        int slower[20];

        int supper[20];

void berth(){

        for(int s = 1; s < 73; s++){

              if (s % 8 == 1 || s % 8 == 4){

                                lower[a] = s;

                                a++;

              }

              else if (s % 8 == 2 || s % 8 == 5){

                                middle[b] = s;

                                b++;

              }

              else if(s % 8 == 3 || s % 8 == 6){

                                upper[c] = s;

                                c++;

              }

              else if(s % 8 == 7){

                                slower[d] = s;

                                d++;

              }

              else{

                                supper[e] = s;

                                e++;

              }

   }

}


int ptr1=0, ptr2=0,ptr3=0,ptr4=0,ptr5=0;

int c1=0,c2=0,c3=0,c4=0,c5=0;
```

```
void bookagain(alldetails* obj);
void bookseat(int num[],string berth,alldetails*obj){
    int k=0;
    for(int i=1;i<73;i++){
        if(num[k]==i){
            if(arr[i]==NULL){
                obj->seatnumber=num[i];
                obj->berth_info=berth;
                break;
            }
            k++;
        }
    }
}

void booklower(alldetails* obj){
    if(ptr1<a){
        obj->seatnumber=lower[ptr1];
        obj->berth_info="lower";
        cout<<"Seat number "<<lower[ptr1]<<" is booked\n";
        ptr1++;
        c1++;
    }else{
        if(c1<a){
            bookseat(lower,"lower",obj);
            return;
        }
        cout<<"No lower available book other berth\n";
        bookagain(obj);
    }
}
void bookmiddle(alldetails* obj){
    if(ptr2<b){
        obj->seatnumber=middle[ptr2];
        obj->berth_info="middle";
        cout<<"Seat number "<<middle[ptr2]<<" is booked\n";
        ptr2++;
        c2++;
    }
else{
        if(c2<b){
            bookseat(middle,"middle",obj);
            return;
        }
        cout<<"No middle available book another berth\n";
        bookagain(obj);
    }
}
void bookupper(alldetails* obj){
    if(ptr3<c){
        obj->seatnumber=upper[ptr3];
        obj->berth_info="upper";
        cout<<"Seat number "<<upper[ptr3]<<" is booked\n";
        ptr3++;
        c3++;
    }else{
        if(c3<b){
            bookseat(upper,"upper",obj);
				return;
			}
			cout<<"No upper available book another berth\n";
			bookagain(obj);
		}
}

void bookslower(alldetails* obj){
		if(ptr4<d){
			obj->seatnumber=slower[ptr4];
			obj->berth_info="sidelower";
			cout<<"Seat number "<<slower[ptr4]<<" is booked\n";
			ptr4++;
			c4++;
		}
		else{
			if(c4<d){
				bookseat(slower,"sidelower",obj);
				return;
			}
			cout<<"No side lower available book another berth\n";
			bookagain(obj);
		}
}
void booksupper(alldetails* obj){
		if(ptr5<e){
			obj->seatnumber=supper[ptr5];
			obj->berth_info="sideupper";
			cout<<"Seat number "<<supper[ptr5]<<" is booked\n";
```

```
                ptr5++;
                c5++;
            }
            else{
                if(c5<e){
                    bookseat(supper,"sideupper",obj);
                    return;
                }
                cout<<"No side upper available book another berth\n";
                bookagain(obj);
            }
}
void bookrandom(alldetails* obj){
            for(int i=1;i<74;i++){
                if(arr[i]==NULL){
                    obj->seatnumber=i;
                    if(i % 8 == 1 || i % 8 == 4){
                        obj->berth_info="lower";
                        cout<<"Seat number "<<i<<" is booked\n";
                        break;
                    }
                    else if(i % 8 == 2 || i % 8 == 5){
                        obj->berth_info="middle";
                        cout<<"Seat number "<<i<<" is booked\n";
                        break;
                    }
                    else if(i % 8 == 3 || i % 8 == 6){
                        obj->berth_info="upper";
                        cout<<"Seat number "<<i<<" is booked\n";
                        break;
                    }
                    else if(i % 8 == 7){
                        obj->berth_info="sidelower";
                        cout<<"Seat number "<<i<<" is booked\n";
                        break;
                    }
                    else{
                        obj->berth_info="sideupper";
                        cout<<"Seat number "<<i<<" is booked\n";
                        break;
                    }
                }
        }
}
void bookagain(alldetails* obj){
        int k;
        cout << "\nBerth Preference:\n \t 1 Lower berth\n \t 2 Middle berth \n \t 3 Upper
berth\n \t 4 Side upper berth\n \t 5 Side lower berth\n \t 6 No preference\n";
        cin >> k;
        switch(k)
        {
        case 1: booklower(obj); break;
        case 2: bookmiddle(obj); break;
        case 3: bookupper(obj); break;
        case 4: booksupper(obj); break;
        case 5: bookslower(obj); break;
        case 6: bookrandom(obj); break;
        }
```

```
}
void reserve(){
	alldetails* obj=new alldetails();
	cout << "Enter your details";
	cout << "\nName: ";
	cin >> obj->name_ls;
	cout << "\nAge: ";
	cin >> obj->age;
	cout << "\nGender(M/F/O): ";
	cin >> obj->gender;
	int k;
	cout << "\nBerth Preference:\n \t 1 Lower berth\n \t 2 Middle berth \n \t 3 Upper 
berth\n \t 4 Side upper berth\n \t 5 Side lower berth\n \t 6 No preference\n";
	cin >> k;
	switch(k)
	{
	case 1: booklower(obj); break;
	case 2: bookmiddle(obj); break;
	case 3: bookupper(obj); break;
	case 4: booksupper(obj); break;
	case 5: bookslower(obj); break;
	case 6: bookrandom(obj); break;
	}
	arr[obj->seatnumber]=obj;
}
void cancel(){
	int s;
	cout << "Enter your seat number\n";
	cin >> s;
```

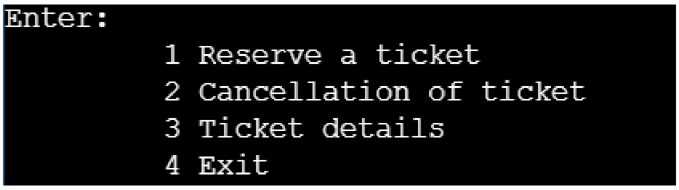
```
        if(arr[s]==NULL)
                cout<<"No existing reservation \n";
        if(s % 8 == 1 || s % 8 == 4)
                c1--;
        else if(s % 8 == 2 || s % 8 == 5)
                c2--;
        else if(s % 8 == 3 || s % 8 == 6)
                c3--;
        else if(s % 8 == 7)
                c4--;
        else
                c5--;
        arr[s]=NULL;
}
void details(){
        int number;
        cout << "Enter your seat number\n";
        cin >> number;
        alldetails* temp=arr[number];
        if(temp==NULL){
                cout<<"Invalid booking\n";
                return;
        }

        cout<<"\nName: "<<temp->name_ls<<"\nAge: "<<temp->age<<"\nGender: "<<temp->gender<<"\nSeat number: "<<temp->seatnumber<<"\nBerth: "<<temp->berth_info<<endl;
}

int main(){
        berth();
        int k=0;
        while (k!=4){
        cout << "\nEnter:\n \t 1 Reserve a ticket\n \t 2 Cancellation of ticket \n \t 3 Ticket details\n \t 4 Exit\n";
        cin >> k;
        switch(k)
        {
        case 1: reserve(); break;
        case 2: cancel(); break;
        case 3: details(); break;
        }
}
cout<<"Exiting\n";
}
```

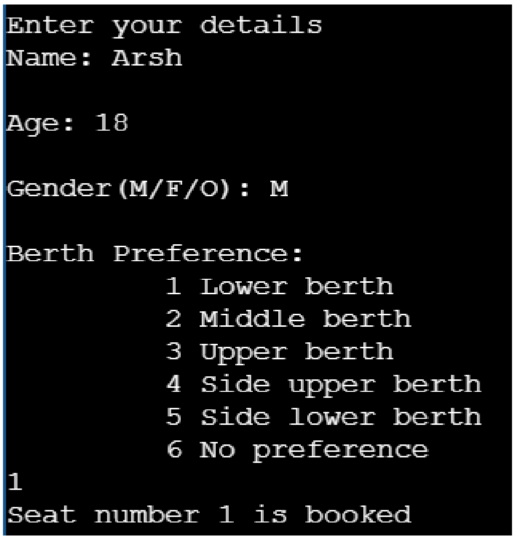
INPUTS AND OUTPUTS

1. First Visible screen



```
Enter:
         1 Reserve a ticket
         2 Cancellation of ticket
         3 Ticket details
         4 Exit
```

2. Reservation of seat



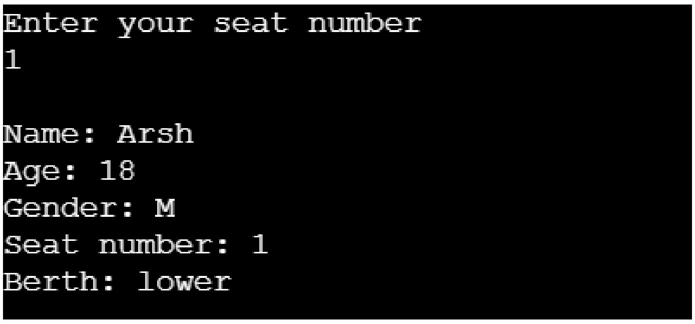
```
Enter your details
Name: Arsh

Age: 18

Gender(M/F/O): M

Berth Preference:
         1 Lower berth
         2 Middle berth
         3 Upper berth
         4 Side upper berth
         5 Side lower berth
         6 No preference
1
Seat number 1 is booked
```

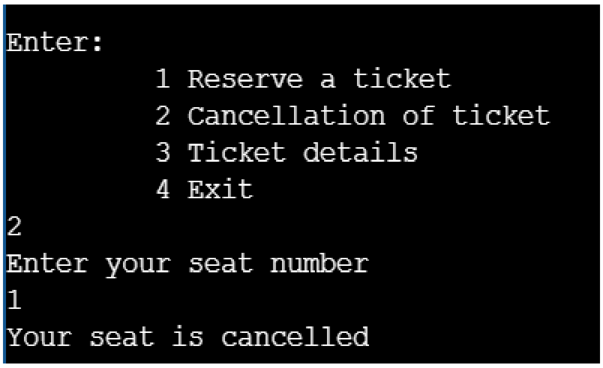
3. Details of Seat Holder



```
Enter your seat number
1

Name: Arsh
Age: 18
Gender: M
Seat number: 1
Berth: lower
```

4. Seat Cancellation



```
Enter:
         1 Reserve a ticket
         2 Cancellation of ticket
         3 Ticket details
         4 Exit
2
Enter your seat number
1
Your seat is cancelled
```