Bus Seat Booking System - Code Explanation

Header Files and Constants

```
#include <stdio.h>
#include <stdlib.h>
#define ROWS 5
#define COLS 4
```

- `stdio.h` is included for input/output functions.
- `stdlib.h` is included for general functions like file handling.
- `ROWS` and `COLS` define the number of rows and columns (seats) in the bus.

Seat Array

```
char seats[ROWS][COLS];
```

- This 2D array represents the seats. Each seat is either 'A' (Available) or 'B' (Booked).

Functions

1. **initializeSeats()**

```
void initializeSeats() {
    for(int i =0; i < ROWS; i++)
        for(int j =0; j < COLS; j++)
            seats[i][j] = 'A';
}
```

- Initializes all seats to 'A' (Available)..

2. **displaySeats()**

```
void displaySeats() {
    printf("\nSeat Map of the bus shown below !!\n(A=Available,
B=Booked):\n\n");
    printf("      ");
    for (int j = 0; j < COLS; j++) printf("  %d ", j);
    printf("\n");
    for (int i = 0; i < ROWS; i++) {
        printf("Row %d:", i);
        for (int j = 1; j <= COLS; j++) {
            printf(" [%c]", seats[i][j]);
        }
        printf("\n");
    }
}
```

- Displays seat arrangement with rows and columns.

3. **saveBookingsToFile()**

```
void saveBookingsToFile() {
    FILE *fp = fopen("bookings.txt", "w");
    if (fp == NULL) {
        printf("Error opening file.\n");
        return;
    }
    for(int i = 0; i < ROWS; i++) {
        for(int j = 0; j < COLS; j++) {
            fprintf(fp, "%c ", seats[i][j]);
        }
        fprintf(fp, "\n");
    }
    fclose(fp);
    printf("Bookings saved to file.\n");
}
```

- Saves the current booking status to a file named `bookings.txt`.

4. **bookSeat(int row, int col)**

```
void bookSeat(int row,int col) {
    if (row > ROWS || col > COLS || row < 0 || col < 0) {
        printf("Invalid seat position.\n");
        return;
    }
    if (seats[row][col] == 'B') {
        printf("Seat already booked.\n");
    } else {
        seats[row][col] = 'B';
        printf("Seat booked successfully.\n");
    }
}
```

- Books a seat if available. Otherwise, informs if already booked or invalid.

5. **cancelSeat(int row, int col)**

```
void cancelSeat(int row, int col) {
    if (row > ROWS || col > COLS || row < 0 || col < 0) {
        printf("Invalid seat position.\n");
        return;
    }
    if (seats[row][col] == 'A') {
        printf("Seat is already available.\n");
    } else {
        seats[row][col] = 'A';
        printf("Seat booking cancelled.\n");
    }
}
```

- Cancels a booked seat if it is already booked.

Main Function

```
int main() {
    int choice, row, col;
    initializeSeats();
    printf("BUS SEAT BOOKING SYSTEM\n");
    while (1) {
        displaySeats();
        printf("\n1. Book Seat\n2. Cancel Seat\n3. Save to File\n4.
Exit\nEnter your choice: ");
        scanf("%d", &choice);
        switch (choice) {
            case 1:
                printf("Enter row to book (1-%d): ", ROWS);
                scanf("%d", &row);
                printf("Enter col to book (1-%d):", COLS);
                scanf("%d", &col);
                bookSeat(row-1, col-1);
                break;
            case 2:
                printf("Enter row to cancel (1-%d): ", ROWS);
                scanf("%d", &row);
                printf("Enter column to cancel (1-%d):", COLS);
                scanf("%d", &col);
                cancelSeat(row-1, col-1);
                break;
            case 3:
                saveBookingsToFile();
                break;
            case 4:
                printf("Thanks for visiting our panel");
                printf("Exiting...\n");
                return 0;
            default:
                printf("Invalid choice.\n");
        }
    }
    return 0;
}
```

- Main driver of the program. Shows menu, takes input, and calls appropriate functions.

Example Walkthrough:

1. Program Starts -> All seats are Available.
2. User chooses Option 1 (Book Seat)
    - Inputs row=2, col=3
    - Seat (2,3) is booked.
3. User chooses Option 1 again
    - Inputs row=2, col=3
    - Program says "Seat already booked."

4. User chooses Option 2 (Cancel Seat)
    - Inputs row=2, col=3
    - Seat booking cancelled (Seat becomes available again).
5. User chooses Option 3 (Save to File)
    - Booking details saved to `bookings.txt` file.
6. User chooses Option 4 (Exit)
    - Program ends.

Source code:

```c
#include <stdio.h>
#include <stdlib.h>

#define ROWS 5
#define COLS 4
char seats[ROWS][COLS];

void initializeSeats() {
    for(int i = 1; i <=ROWS; i++)
        for(int j = 1; j <= COLS; j++)
            seats[i][j] = 'A';
}

void displaySeats() {
    printf("\nSeat Map of the bus shown below !!\n(A=Available, B=Booked):\n\n");

    printf("      ");
    for (int j = 1; j <= COLS; j++) {
        printf("  %d ", j);
    }
    printf("\n");

    for (int i = 1; i <= ROWS; i++) {
        printf("Row %d:", i);
        for (int j = 1; j <= COLS; j++) {
            printf(" [%c]", seats[i][j]);
        }
        printf("\n");
    }
}



void saveBookingsToFile() {
    FILE *fp = fopen("bookings.txt", "w");
    if (fp == NULL) {
        printf("Error opening file.\n");
        return;
    }
    for(int i = 1; i <=ROWS; i++) {
        for(int j = 1; j <= COLS; j++) {
            fprintf(fp, "%c ", seats[i][j]);
        }
        fprintf(fp, "\n");
    }
    fclose(fp);
    printf("Bookings saved to file.\n");
}

void bookSeat(int row,int col) {
    if (row >ROWS || col >COLS || row < 0 || col < 0) {
        printf("Invalid seat position.\n");
        return;
    }
    if (seats[row][col] == 'B') {
        printf("Seat already booked.\n");
    } else {
        seats[row][col] = 'B';
        printf("Seat booked successfully.\n");
    }
}

void cancelSeat(int row, int col) {
    if (row > ROWS || col > COLS || row < 0 || col < 0) {
        printf("Invalid seat position.\n");
        return;
    }
    if (seats[row][col] == 'A') {
        printf("Seat is already available.\n");
    } else {
        seats[row][col] = 'A';
        printf("Seat booking cancelled.\n");
    }
```

```c
}

int main() {
    int choice, row, col;
    initializeSeats();
         printf("BUS SEAT BOOKING SYSTEM\n");
    while (1) {
          displaySeats();
        printf("\n1. Book Seat\n2. Cancel Seat\n3. Save to File\n4. Exit\nEnter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter row  to book (1-%d): ",ROWS);
                scanf("%d", &row);
                printf("Enter col to book (1-%d):",COLS);
                scanf("%d",&col);
                bookSeat(row, col);
                break;
            case 2:
                printf("Enter row to cancel (1-%d): ",ROWS);
                scanf("%d", &row);
                printf("Enter column to cancel (1-%d):",COLS);
                scanf("%d",&col);
                cancelSeat(row, col);
                break;
            case 3:
                saveBookingsToFile();
                break;
            case 4:
                  printf("Thanks for visiting our panel");
                printf("Exiting...\n");
                return 0;
            default:
                printf("Invalid choice.\n");
        }
    }
    return 0;
}
```

Output:

```
       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [A] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice:
```

```
       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [A] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice: 1
Enter row  to book (1-5): 2
Enter col to book (1-4):3
Seat booked successfully.

Seat Map of the bus shown below !!
(A=Available, B=Booked):

       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [B] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice:
```

Ln 111, Col 2 (2970 selected) Spaces: 4 UTF-8 CRLF C

```
Enter col to book (1-4):3
Seat booked successfully.

Seat Map of the bus shown below !!
(A=Available, B=Booked):

       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [B] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice: 2
Enter row to cancel (1-5): 2
Enter column to cancel (1-4):3
Seat booking cancelled.

Seat Map of the bus shown below !!
(A=Available, B=Booked):

       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [A] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice:
```

```
Seat Map of the bus shown below !!
(A=Available, B=Booked):

       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [B] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice: 2
Enter row to cancel (1-5): 2
Enter column to cancel (1-4):3
Seat booking cancelled.

Seat Map of the bus shown below !!
(A=Available, B=Booked):

       1   2   3   4
Row 1: [A] [A] [A] [A]
Row 2: [A] [A] [A] [A]
Row 3: [A] [A] [A] [A]
Row 4: [A] [A] [A] [A]
Row 5: [A] [A] [A] [A]

1. Book Seat
2. Cancel Seat
3. Save to File
4. Exit
Enter your choice: 4
Thanks for visiting our panelExiting...
PS C:\c program 2025>
```