

//simple C++ program that demonstrates how to add,
//delete, and search for an element in a 2D array.

```
#include <iostream>
```

```
using namespace std;
```

```
const int ROWS = 3;
```

```
const int COLS = 3;
```

```
// Function to display the 2D array
```

```
void displayArray(int arr[ROWS][COLS]) {
```

```
    for (int i = 0; i < ROWS; ++i) {
```

```
        for (int j = 0; j < COLS; ++j) {
```

```
            cout << arr[i][j] << " ";
```

```
        }
```

```
        cout << endl;
```

```
    }
```

```
    cout << endl;
```

```
}
```

// Function to add an element to the 2D array at a specific position

```
void addElement(int arr[ROWS][COLS], int row, int col,
int value) {
    if (row >= 0 && row < ROWS && col >= 0 && col <
COLS) {
        arr[row][col] = value;
        cout << "Element added successfully." << endl;
    } else {
        cout << "Invalid position!" << endl;
    }
}
```

// Function to delete an element from the 2D array by setting it to 0 (or another placeholder value)

```
void deleteElement(int arr[ROWS][COLS], int row, int col)
{
    if (row >= 0 && row < ROWS && col >= 0 && col <
COLS) {
        arr[row][col] = 0; // Set to 0 to indicate deletion
        cout << "Element deleted successfully." << endl;
    } else {
        cout << "Invalid position!" << endl;
    }
}
```

```

    }
}

// Function to search for an element in the 2D array
bool searchElement(int arr[ROWS][COLS], int value) {
    for (int i = 0; i < ROWS; ++i) {
        for (int j = 0; j < COLS; ++j) {
            if (arr[i][j] == value) {
                cout << "Element found at position (" << i << ", "
<< j << ")" << endl;
                return true;
            }
        }
    }
    cout << "Element not found." << endl;
    return false;
}

```

```
int main() {  
    int array[ROWS][COLS] = { {1, 2, 3}, {4, 5, 6}, {7, 8,  
9} };  
  
    cout << "Initial Array:" << endl;  
    displayArray(array);  
  
    // Add an element  
    addElement(array, 1, 1, 10);  
    cout << "Array after adding element:" << endl;  
    displayArray(array);  
  
    // Delete an element  
    deleteElement(array, 2, 2);  
    cout << "Array after deleting element:" << endl;  
    displayArray(array);  
  
    // Search for an element  
    searchElement(array, 10);  
    return 0;  
}
```