

Name : Kishan Vaghamashi

Student ID: 202312014

1)

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;

int main()
{
    ll n1 = 0b10111001;
    ll n2 = 0b10010110;
    cout << "Decimal = " << (n1 & n2) << "\n";
    cout << "Binary = " << bitset<8>(n1 & n2) << "\n";

    n1 = 0b11011110;
    n2 = 0b11000101;
    cout << "Decimal = " << (n1 & n2) << "\n";
    cout << "Binary = " << bitset<8>(n1 & n2) << "\n";

    n1 = 0b01111101;
    n2 = 0b10111110;
    cout << "Decimal = " << (n1 | n2) << "\n";
    cout << "Binary = " << bitset<8>(n1 | n2) << "\n";

    n1 = 0b11000110;
    n2 = 0b11011100;
    cout << "Decimal = " << (n1 | n2) << "\n";
    cout << "Binary = " << bitset<8>(n1 | n2) << "\n";

    n1 = 0b10111001;
    n2 = 0b11110110;
    cout << "Decimal = " << (n1 ^ n2) << "\n";
    cout << "Binary = " << bitset<8>(n1 ^ n2) << "\n";

    n1 = 0b11000010;
    n2 = 0b00000101;
    cout << "Decimal = " << (n1 ^ n2) << "\n";
    cout << "Binary = " << bitset<8>(n1 ^ n2) << "\n";

    n1 = 0b1011100110010110;
    cout << "Binary = " << bitset<16>(~n1) << "\n";

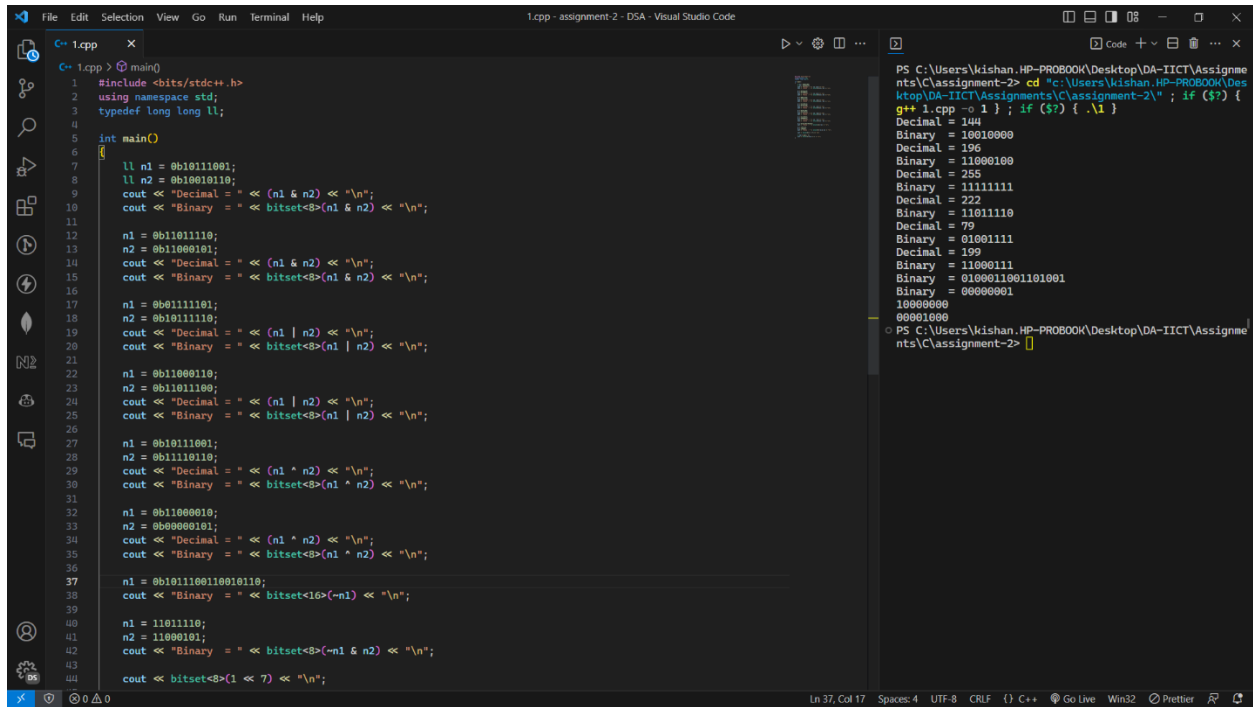
    n1 = 11011110;
    n2 = 11000101;
    cout << "Binary = " << bitset<8>(~n1 & n2) << "\n";
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
cout << bitset<8>(1 << 7) << "\n";

ll num = pow(2, 7);
cout << bitset<8>(num >> 4) << "\n";
}
```



```
C++ 1.cpp > main()
1 #include <bits/stdc++.h>
2 using namespace std;
3 typedef long long ll;
4
5 int main()
6 {
7     ll n1 = 0b10111001;
8     ll n2 = 0b10010110;
9     cout << "Decimal = " << (n1 & n2) << "\n";
10    cout << "Binary = " << bitset<8>(n1 & n2) << "\n";
11
12    n1 = 0b11011110;
13    n2 = 0b11000101;
14    cout << "Decimal = " << (n1 & n2) << "\n";
15    cout << "Binary = " << bitset<8>(n1 & n2) << "\n";
16
17    n1 = 0b01111101;
18    n2 = 0b10111110;
19    cout << "Decimal = " << (n1 | n2) << "\n";
20    cout << "Binary = " << bitset<8>(n1 | n2) << "\n";
21
22    n1 = 0b11000110;
23    n2 = 0b11011100;
24    cout << "Decimal = " << (n1 | n2) << "\n";
25    cout << "Binary = " << bitset<8>(n1 | n2) << "\n";
26
27    n1 = 0b10111001;
28    n2 = 0b11101110;
29    cout << "Decimal = " << (n1 ^ n2) << "\n";
30    cout << "Binary = " << bitset<8>(n1 ^ n2) << "\n";
31
32    n1 = 0b11000010;
33    n2 = 0b00000101;
34    cout << "Decimal = " << (n1 ^ n2) << "\n";
35    cout << "Binary = " << bitset<8>(n1 ^ n2) << "\n";
36
37    n1 = 0b1011100110010110;
38    cout << "Binary = " << bitset<16>(~n1) << "\n";
39
40    n1 = 11011110;
41    n2 = 11000101;
42    cout << "Binary = " << bitset<8>(~n1 & n2) << "\n";
43
44    cout << bitset<8>(1 << 7) << "\n";
45 }
```

PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2> cd "c:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2" ; if (\$?) { g++ 1.cpp -o 1 ; if (\$?) { .\1 } }
Decimal = 144
Binary = 10010000
Decimal = 196
Binary = 11000100
Decimal = 255
Binary = 11111111
Decimal = 222
Binary = 11011110
Decimal = 79
Binary = 01001111
Decimal = 199
Binary = 11000111
Binary = 010001100101001
Binary = 00000001
10000000
00001000
PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2>

2)

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;

int main()
{
    ll bit = 0b10101110101010110101110100111011;

    cout << "1)\n";
    // set 16th bit
    bitset<32> result(bit | (1 << 15));
    cout << result << "\n";

    cout << "2)\n";
    // set 28th bit
    result = bitset<32>(bit & ~(1 << 27));
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
cout << result << "\n";

bitset<16> oddBit;
bitset<16> evenBit;
ll temp{bit};

int i = 0;
while (temp << 1 != 0)
{
    oddBit[i++ / 2] = (temp & 1);
    temp = temp >> 1;
    evenBit[i++ / 2] = (temp & 1);
    temp = temp >> 1;
}

cout << "3)\n";
cout << "Odd Bit : " << oddBit << "\n";
cout << "Even Bit : " << evenBit << "\n";

cout << "4)\n";
bitset<32> tempBit(bit);
int n = 3; // 3rd nibble
for (size_t i = 4 * (n - 1); i < 4 * n; i++)
{
    tempBit[i] = ~tempBit[i];
}

cout << tempBit << "\n";
return 0;
}
```

The screenshot shows the Visual Studio Code editor with a C++ file named '2.cpp'. The code is identical to the one in the previous block. The output window on the right shows the following results:

```
PS C:\Users\Kishan_HP-PROBOOK\Desktop\DA
-IICT\Assignments\C\Assignment-2> cd "c:\
Users\Kishan_HP-PROBOOK\Desktop\DA-IICT
\Assignments\C\Assignment-2\" ; if ($?)
{ g++ 2.cpp -o 2 ; if ($?) { .\2 }
1)
101011101010101110101110100111011
2)
101001101010101010101110100111011
3)
Odd Bit : 0010000111110101
Even Bit : 1111111100100111
4)
101011101010101010101001000111011
PS C:\Users\Kishan_HP-PROBOOK\Desktop\DA
-IICT\Assignments\C\Assignment-2>
```

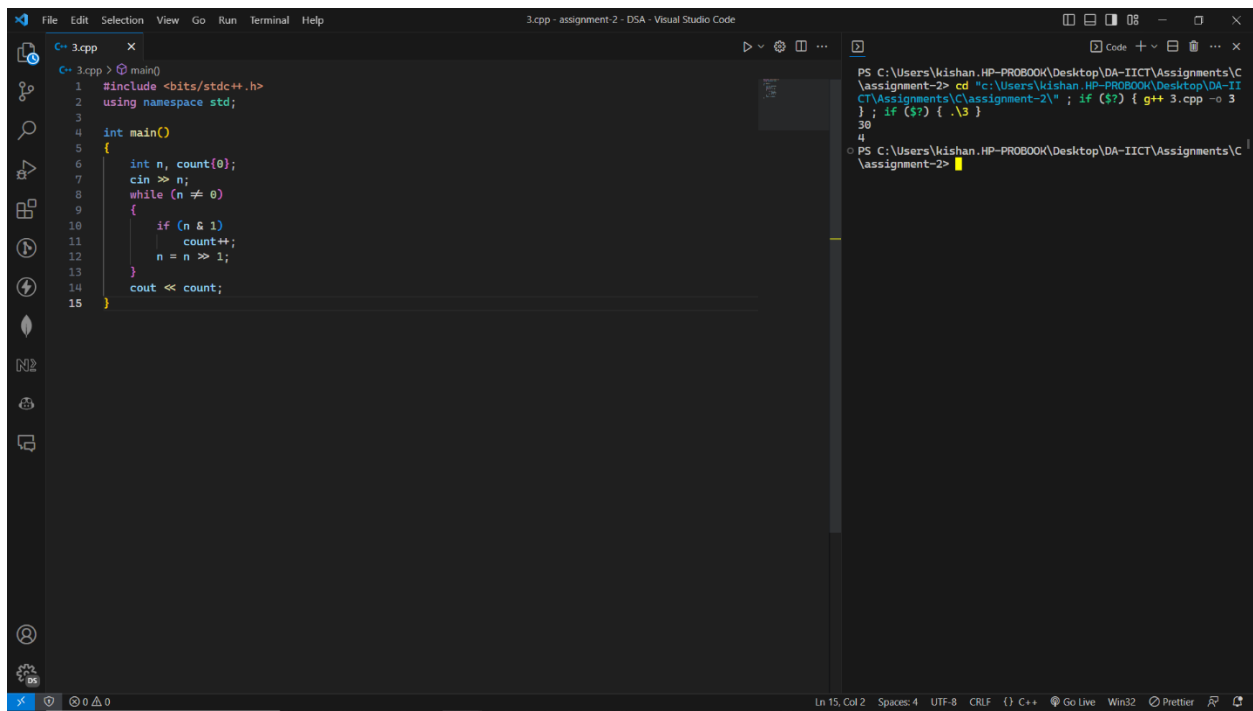
Name : Kishan Vaghamashi

Student ID: 202312014

3)

```
#include <bits/stdc++.h>
using namespace std;

int main()
{
    int n, count{0};
    cin >> n;
    while (n != 0)
    {
        if (n & 1)
            count++;
        n = n >> 1;
    }
    cout << count;
}
```



4)

```
#include <bits/stdc++.h>
using namespace std;

// 0 - Tablet Mode
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
// 1 - WiFi
// 2 - Mute
// 3 - Airplane Mode
// 4 - Auto Hide Taskbar

const vector<string> settingNames = {
    "Tablet Mode",
    "WiFi",
    "Mute",
    "Airplane Mode",
    "Auto Hide Taskbar"};

void display(bitset<5> &settings)
{
    cout << "\n=====\\n";
    for (size_t i = 0; i < settings.size(); i++)
    {
        if (settings[i])
        {
            cout << settingNames[i];
            cout << " is *ON*\\n";
        }
        else
        {
            cout << settingNames[i];
            cout << " is OFF\\n";
        }
    }
    cout << "=====\\n";
}

void takeInput(bitset<5> &settings)
{
    cout << "\\n=>Select option to toggle<=\\n";
    for (size_t i = 0; i < settingNames.size(); i++)
    {
        cout << i + 1 << ") " << settingNames[i] << "\\n";
    }
    int input;
    cin >> input;

    if (input >= 1 && input <= 5)
    {
        settings[input - 1] = ~settings[input - 1];
    }
}
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
    else
    {
        cout << "Invalid input\n";
    }
    display(settings);
}

int main()
{
    bitset<5> settings(00000);
    int input;
    bool isExist = false;
    display(settings);
    while (1)
    {
        if (isExist)
            break;

        cout << "Select one of the option...\n";
        cout << "1) Display the settings\n";
        cout << "2) Change the settings\n";
        cout << "3) EXIT\n";
        cin >> input;

        switch (input)
        {
            case 1:
                display(settings);
                break;

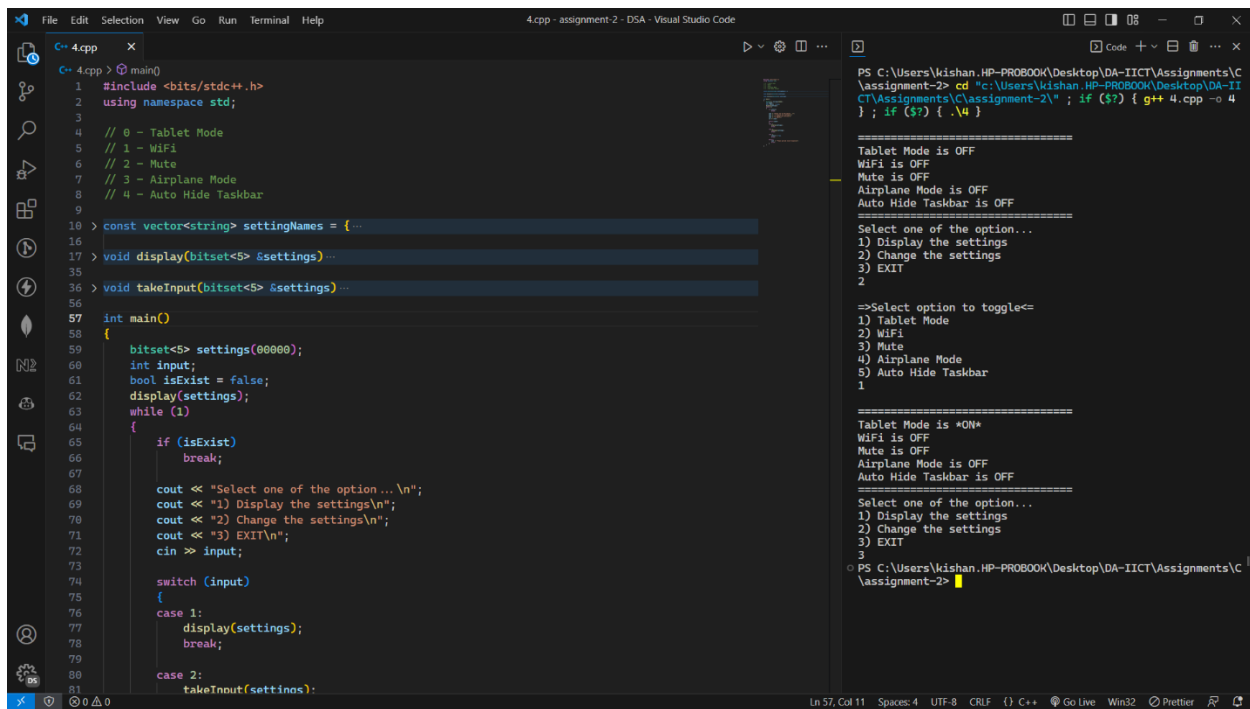
            case 2:
                takeInput(settings);
                break;

            case 3:
                isExist = true;
                break;

            default:
                cout << "Please provide valid response\n";
                break;
        }
    }
}
```

Name : Kishan Vaghamashi

Student ID: 202312014



The screenshot shows a Visual Studio Code editor with a C++ file named 4.cpp. The code implements a settings application using a bitset to manage five settings: Tablet Mode, WiFi, Mute, Airplane Mode, and Auto Hide Taskbar. The main function initializes the settings, displays them, and enters a loop where the user can toggle settings or exit. The terminal output shows the program's execution, displaying the initial settings and the user's interactions.

```
#include <bits/stdc++.h>
using namespace std;

// 0 - Tablet Mode
// 1 - WiFi
// 2 - Mute
// 3 - Airplane Mode
// 4 - Auto Hide Taskbar

const vector<string> settingNames = { ... };

void display(bitset<5> &settings) ...

void takeInput(bitset<5> &settings) ...

int main()
{
    bitset<5> settings(00000);
    int input;
    bool isExist = false;
    display(settings);
    while (1)
    {
        if (!isExist)
            break;

        cout << "Select one of the option...\n";
        cout << "1) Display the settings\n";
        cout << "2) Change the settings\n";
        cout << "3) EXIT\n";
        cin >> input;

        switch (input)
        {
            case 1:
                display(settings);
                break;
            case 2:
                takeInput(settings);
                break;
            case 3:
                return 0;
        }
    }
}
```

Terminal Output:

```
PS C:\Users\Kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\C\Assignment-2> cd "c:\Users\Kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\C\Assignment-2\" ; if ($?) { g++ 4.cpp -o 4 } ; if ($?) { .\4 }

=====
Tablet Mode is OFF
WiFi is OFF
Mute is OFF
Airplane Mode is OFF
Auto Hide Taskbar is OFF
=====
Select one of the option...
1) Display the settings
2) Change the settings
3) EXIT
2

=>Select option to toggle<=
1) Tablet Mode
2) WiFi
3) Mute
4) Airplane Mode
5) Auto Hide Taskbar
1

=====
Tablet Mode is *ON*
WiFi is OFF
Mute is OFF
Airplane Mode is OFF
Auto Hide Taskbar is OFF
=====
Select one of the option...
1) Display the settings
2) Change the settings
3) EXIT
3
PS C:\Users\Kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\C\Assignment-2>
```

5)

```
#include <iostream>
using namespace std;
typedef unsigned int uint;
void print(int num)
{
    cout << "\nValues Selected : \n\n";
    int temp = 0;
    while (temp < 12)
    {
        if ((temp + 1) == 1)
        {
            if (num & 1)
                cout << "Seat Covers";
        }
        else if ((temp + 1) == 2)
        {
            if (num & 1)
                cout << " : Beige ";
        }
        else if ((temp + 1) == 3)
        {
            if (num & 1)
                cout << " : Dark ";
        }
        temp++;
    }
}
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
}
else if ((temp + 1) == 4)
{
    if (num & 1)
        cout << " : Dual Pattern ";
}
else if ((temp + 1) == 5)
{
    if (num & 1)
        cout << "\n"
            << "Alloys";
}
else if ((temp + 1) == 6)
{
    if (num & 1)
        cout << "\n"
            << "Colors ";
}
else if ((temp + 1) == 7)
{
    if (num & 1)
        cout << " : Coffee Brown ";
}
else if ((temp + 1) == 8)
{
    if (num & 1)
        cout << " : Pearl White";
}
else if ((temp + 1) == 9)
{
    if (num & 1)
        cout << " : Marine Blue ";
}
else if ((temp + 1) == 10)
{
    if (num & 1)
        cout << " : Ash Grey ";
}
else if ((temp + 1) == 11)
{
    if (num & 1)
        cout << "\n"
            << "Steering Cover";
}
else if ((temp + 1) == 12)
```


Name : Kishan Vaghamashi

Student ID: 202312014

```
{
    if (num & 1)
        cout << "\n"
              << "Body Cover"
              << "\n";
    }
    num = num >> 1;
    temp++;
}
cout << "\n\n";
}
void CarCompany()
{
    unsigned int num{0b0}, bcv = {0b1};
    bool select = 0;
    int ch;
    cout << "\n--- Welcome To Car Company---\n\n";
    cout << "Select for \"Seat Cover\" (0 for no / 1 for yes): ";
    cin >> select;
    if (select == 1)
    {
        num = num | bcv;
        cout << "\nSelect for Seat cover :- \n";
        cout << " 1) Beige\n";
        cout << " 2) Dark\n";
        cout << " 3) Dual Pattern\n";
        cout << "Enter Choice : ";
        cin >> ch;
        switch (ch)
        {
            case 1:
                num = num | bcv << 1;
                break;
            case 2:
                num = num | bcv << 2;
                break;
            case 3:
                num = num | bcv << 3;
                break;
            default:
                break;
        }
    }
    cout << "\nDo you want \"Alloys\" (0 for no / 1 for yes): ";
    cin >> select;
```

Name : Kishan Vaghamashi

Student ID: 202312014

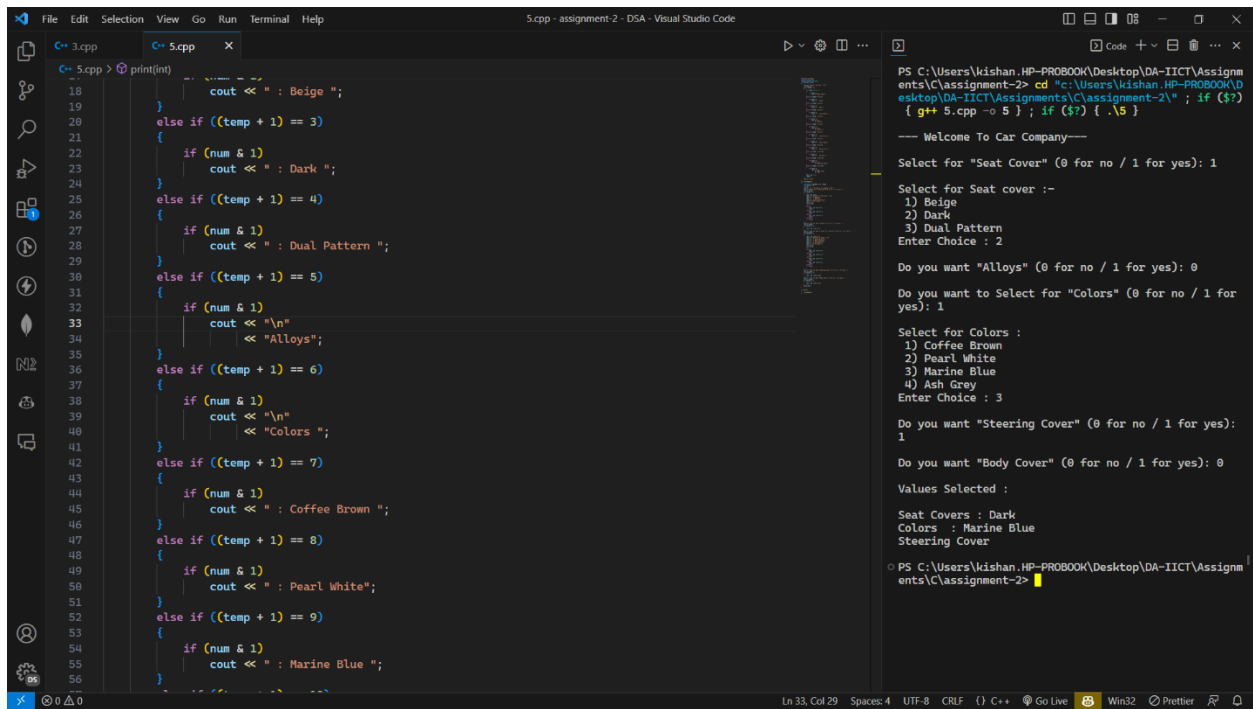
```
if (select == 1)
{
    num = num | bcv << 4;
}
cout << "\nDo you want to Select for \"Colors\" (0 for no / 1 for yes): ";
";
cin >> select;
if (select == 1)
{
    num = num | bcv << 5;
    cout << "\nSelect for Colors : \n";
    cout << " 1) Coffee Brown\n";
    cout << " 2) Pearl White\n";
    cout << " 3) Marine Blue\n";
    cout << " 4) Ash Grey\n";
    cout << "Enter Choice : ";
    cin >> ch;
    switch (ch)
    {
        case 1:
            num = num | bcv << 6;
            break;
        case 2:
            num = num | bcv << 7;
            break;
        case 3:
            num = num | bcv << 8;
            break;
        case 4:
            num = num | bcv << 9;
            break;
        default:
            break;
    }
}
cout << "\nDo you want \"Steering Cover\" (0 for no / 1 for yes): ";
cin >> select;
if (select == 1)
{
    num = num | bcv << 10;
}
cout << "\nDo you want \"Body Cover\" (0 for no / 1 for yes): ";
cin >> select;
if (select == 1)
{
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
        num = num | bcv << 11;
    }
    print(num);
}

int main()
{
    CarCompany();
}
```



```
File Edit Selection View Go Run Terminal Help
5.cpp - assignment-2 - DSA - Visual Studio Code

C++ 5.cpp x
5.cpp > print(int)
18         cout << " : Beige ";
19     }
20     else if ((temp + 1) == 3)
21     {
22         if (num & 1)
23             cout << " : Dark ";
24     }
25     else if ((temp + 1) == 4)
26     {
27         if (num & 1)
28             cout << " : Dual Pattern ";
29     }
30     else if ((temp + 1) == 5)
31     {
32         if (num & 1)
33             cout << "\n"
34             << "Alloys";
35     }
36     else if ((temp + 1) == 6)
37     {
38         if (num & 1)
39             cout << "\n"
40             << "Colors ";
41     }
42     else if ((temp + 1) == 7)
43     {
44         if (num & 1)
45             cout << " : Coffee Brown ";
46     }
47     else if ((temp + 1) == 8)
48     {
49         if (num & 1)
50             cout << " : Pearl White";
51     }
52     else if ((temp + 1) == 9)
53     {
54         if (num & 1)
55             cout << " : Marine Blue ";
56     }
57 }

PS C:\Users\Kishan_HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2> cd "C:\Users\Kishan_HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2\"; if ($?) { g++ 5.cpp -o 5 ; if ($?) { .\5 } }

--- Welcome To Car Company---

Select for "Seat Cover" (0 for no / 1 for yes): 1

Select for Seat cover :-
1) Beige
2) Dark
3) Dual Pattern
Enter Choice : 2

Do you want "Alloys" (0 for no / 1 for yes): 0

Do you want to Select for "Colors" (0 for no / 1 for yes): 1

Select for Colors :
1) Coffee Brown
2) Pearl White
3) Marine Blue
4) Ash Grey
Enter Choice : 3

Do you want "Steering Cover" (0 for no / 1 for yes): 1

Do you want "Body Cover" (0 for no / 1 for yes): 0

Values Selected :

Seat Covers : Dark
Colors : Marine Blue
Steering Cover

C:\Users\Kishan_HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2>
```

6)

```
#include <bits/stdc++.h>
using namespace std;

int main()
{
    double arr[5];
    double value = 1.1;

    for (size_t i = 0; i < 5; i++)
    {
        arr[i] = value;
        // cout << value << "\n";
    }
}
```

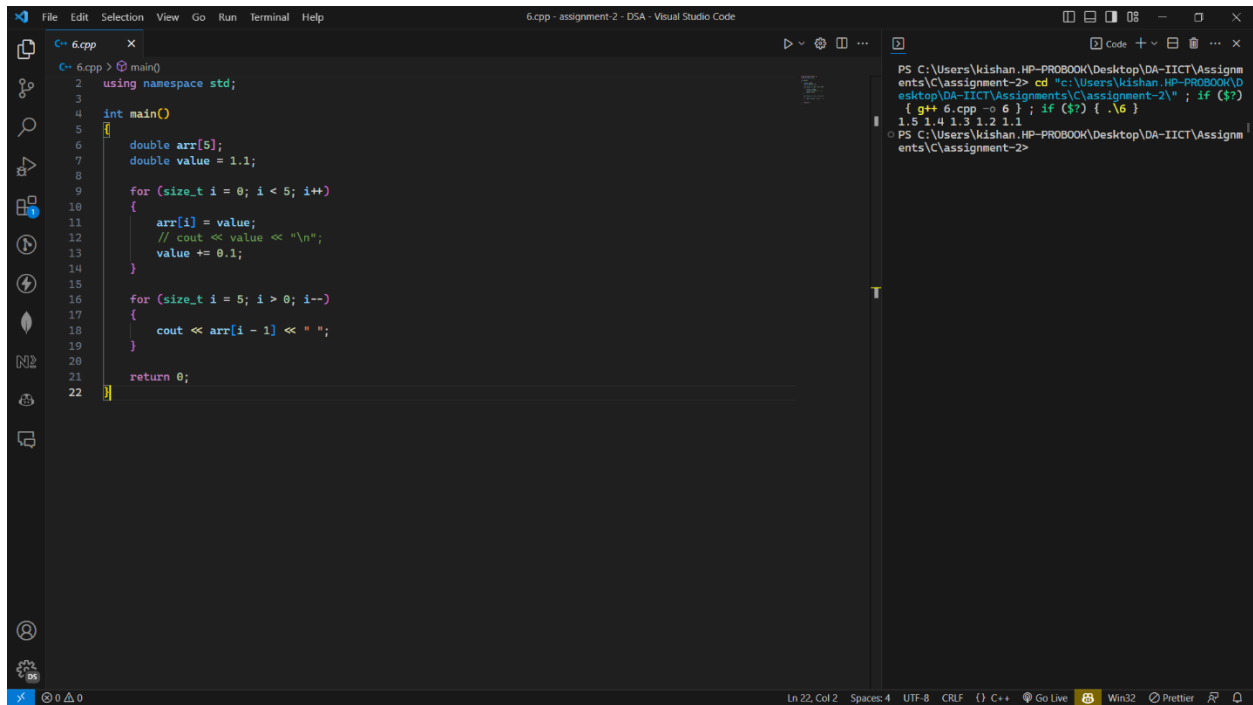
Name : Kishan Vaghamashi

Student ID: 202312014

```
        value += 0.1;
    }

    for (size_t i = 5; i > 0; i--)
    {
        cout << arr[i - 1] << " ";
    }

    return 0;
}
```



The screenshot shows the Visual Studio Code editor with a C++ file named 6.cpp. The code defines an array of 5 doubles, increments a value from 1.1 to 1.5, and then prints the array elements in reverse order. The output window on the right shows the execution results.

```
6.cpp - assignment-2 - DSA - Visual Studio Code

C++ 6.cpp x
1 int main()
2     using namespace std;
3
4     int main()
5     {
6         double arr[5];
7         double value = 1.1;
8
9         for (size_t i = 0; i < 5; i++)
10        {
11            arr[i] = value;
12            // cout << value << "\n";
13            value += 0.1;
14        }
15
16        for (size_t i = 5; i > 0; i--)
17        {
18            cout << arr[i - 1] << " ";
19        }
20
21        return 0;
22    }
```

Output:

```
PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\C\assignment-2> cd "c:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\C\assignment-2" ; if ($?) { g++ 6.cpp -o 6 } ; if ($?) { .\6 }
1.5 1.4 1.3 1.2 1.1
PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\C\assignment-2>
```

7)

```
#include <iostream>
#include <cstring>
#include <string.h>
#include <cctype>

using namespace std;

int main()
{
    char chArrayA[100]{" "}, chArrayB[100]{" "};
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
// a
cin >> chArrayA >> chArrayB;
cout << "a) \n";
cout << "Firstname:- " << chArrayA << "\n"
    << "Lastname :- " << chArrayB << "\n";

// b
strcat(chArrayA, " ");
strcat(chArrayA, chArrayB);
cout << "b) \n";
cout << chArrayA << "\n";

// c
cin.ignore();
cin.getline(chArrayA, 90);
cout << "c) \n";
cout << chArrayA << "\n";

// d
int lastLocA = strlen(chArrayA) - 1;
int lastLocB = 0;
for (int i = lastLocA; i > 0; i--)
{
    chArrayB[lastLocB++] = chArrayA[i];
}
cout << "d) \n";
cout << chArrayB << "\n";

// e
char temp[100][100];
int row{0}, col{0};
temp[0][col++] = ' ';

for (size_t i = 0; i < strlen(chArrayA); i++)
{
    if (chArrayA[i] == ' ')
    {
        row++;
        col = 0;
    }
    temp[row][col++] = chArrayA[i];
}

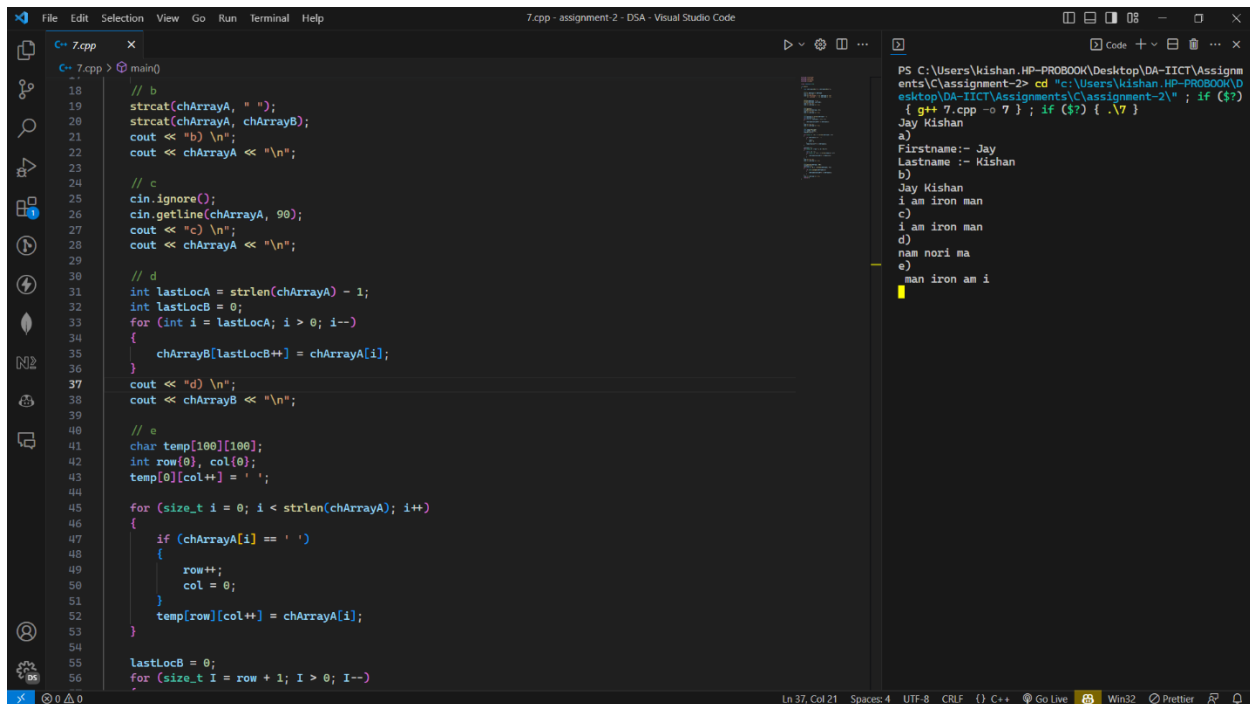
lastLocB = 0;
for (size_t i = row + 1; i > 0; i--)
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
{
    int i = I - 1;
    for (size_t j = 0; j < strlen(temp[i]); j++)
    {
        chArrayB[lastLocB++] = temp[i][j];
    }
}
cout << "e) \n";
cout << chArrayB << "\n";

// f
cin.getline(chArrayA, 100);
lastLocB = 0;
for (int i = 0; i < strlen(chArrayA); ++i)
{
    if (std::isalpha(chArrayA[i]))
    {
        chArrayB[lastLocB++] = chArrayA[i];
    }
}
cout << chArrayB << "\n";
return 0;
}
```



The screenshot shows a Visual Studio Code window with a C++ file named 7.cpp. The code implements a program that processes a string 'Jay Kishan' and prints it in various formats. The code includes comments for each step (a through e) and uses standard C++ libraries like <string>, <string_view>, and <algorithm>. The output window on the right shows the results of the program's execution, which match the expected output from the problem statement.

```
7.cpp - assignment-2 - DSA - Visual Studio Code

C++ 7.cpp x
C++ 7.cpp > main()
18 // b
19 strcat(chArrayA, " ");
20 strcat(chArrayA, chArrayB);
21 cout << "b) \n";
22 cout << chArrayA << "\n";
23
24 // c
25 cin.ignore();
26 cin.getline(chArrayA, 90);
27 cout << "c) \n";
28 cout << chArrayA << "\n";
29
30 // d
31 int lastLocA = strlen(chArrayA) - 1;
32 int lastLocB = 0;
33 for (int i = lastLocA; i > 0; i--)
34 {
35     chArrayB[lastLocB++] = chArrayA[i];
36 }
37 cout << "d) \n";
38 cout << chArrayB << "\n";
39
40 // e
41 char temp[100][100];
42 int row[0], col[0];
43 temp[0][col++] = ' ';
44
45 for (size_t i = 0; i < strlen(chArrayA); i++)
46 {
47     if (chArrayA[i] == ' ')
48     {
49         row++;
50         col = 0;
51     }
52     temp[row][col++] = chArrayA[i];
53 }
54
55 lastLocB = 0;
56 for (size_t I = row + 1; I > 0; I--)
```

PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2> cd "C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2\" ; if (\$?) { g++ 7.cpp -o 7 } ; if (\$?) { . 7 }

Jay Kishan
a)
Firstname:- Jay
Lastname :- Kishan
b)
Jay Kishan
i am iron man
c)
i am iron man
d)
nam nori ma
e)
man iron am i

Name : Kishan Vaghamashi

Student ID: 202312014

8)

```
#include <iostream>
#include <bitset>
using namespace std;

int seats[16];
const int totalRow = 16;
const int totalSeatInRow = 20;
const char seatBooked{'X'};
const char seatAvailale{'.'};

void setInitialData()
{
    for (size_t i = 0; i < totalRow; i++)
    {
        seats[i] = 0;
    }
}

void display()
{
    char rowName = 'A';
    for (size_t i = 0; i < totalRow; i++)
    {
        bitset<20> row(seats[i]);
        for (size_t i = 0; i < row.size(); i++)
        {
            if (row[i] == 1)
            {
                cout << seatBooked;
            }
            else
            {
                cout << seatAvailale;
            }
        }
        cout << "    <=== " << rowName++;
        cout << "\n";
    }
    cout << "VVVVVVVVVVVVVVVVVVVV    Screen this way\n";
}

void buyTicket()
{

```

Name : Kishan Vaghamashi

Student ID: 202312014

```
char rowName{};
int row{0}, col{0};
cout << "Enter row:- ";
cin >> rowName;
cout << "Enter seat number:- ";
cin >> col;
row = (rowName >= 'A' && rowName <= 'P') ? (rowName - 'A') : ((rowName
>= 'a' && rowName <= 'p') ? (rowName - 'a') : -1);

if (row != -1)
{
    if (col > 0 && col <= totalSeatInRow)
    {
        bitset<20> temp = bitset<20>(seats[row]);
        if (temp[col - 1] == 0)
        {
            temp[col - 1] = 1;
            seats[row] = static_cast<int>(temp.to_ulong());
            cout << rowName << "-" << col << " has been booked
successfully.\n";
        }
        else
        {
            cout << "Sorry, this seat is already booked.\n";
        }
    }
    else
    {
        cout << "Seat number you entered is not valid.\n";
    }
}
else
{
    cout << "Please enter vaild seat name.\n";
}
}

void cancelTicket()
{
    char rowName{};
    int row{0}, col{0};
    cout << "Enter row:- ";
    cin >> rowName;
    cout << "Enter seat number:- ";
    cin >> col;
```


Name : Kishan Vaghamashi

Student ID: 202312014

```
        row = (rowName >= 'A' && rowName <= 'P') ? (rowName - 'A') : ((rowName
>= 'a' && rowName <= 'p') ? (rowName - 'a') : -1);

        if (row != -1)
        {
            if (col > 0 && col <= totalSeatInRow)
            {
                bitset<20> temp = bitset<20>(seats[row]);
                if (temp[col - 1] == 1)
                {
                    temp[col - 1] = 0;
                    seats[row] = static_cast<int>(temp.to_ulong());
                    cout << "Your ticket " << rowName << "-" << col << " has
been cancelled successfully.\n";
                }
                else
                {
                    cout << "Please recheck again, this seat is not booked.\n";
                }
            }
            else
            {
                cout << "Seat number you entered is not valid.\n";
            }
        }
        else
        {
            cout << "Please enter vaild row name.\n";
        }
    }

int main()
{
    int choice{0};
    display();
    bool exit = false;
    while (1)
    {
        if (exit)
            break;

        cout << "1) Buy a ticket \n2) Cancel a ticket \n3) Exit";
        cin >> choice;

        switch (choice)
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
{
    case 1:
        cout << "=====\n";
        buyTicket();
        cout << "=====\n";
        display();
        break;

    case 2:
        cout << "=====\n";
        cancelTicket();
        cout << "=====\n";
        display();
        break;

    case 3:
        exit = true;
        break;

    default:
        cout << "Please enter valid choice\n";
        break;
}
}
```

The screenshot shows the Visual Studio Code editor with a C++ file named `8.cpp`. The code defines a program for managing a 16x20 seat grid. It includes functions for setting initial data, displaying the grid, buying tickets, and canceling tickets. The `main` function uses a `switch` statement to handle user input. The output window shows the program's execution, including the initial grid display, a successful ticket purchase for row 5, seat 5, and a subsequent display of the updated grid.

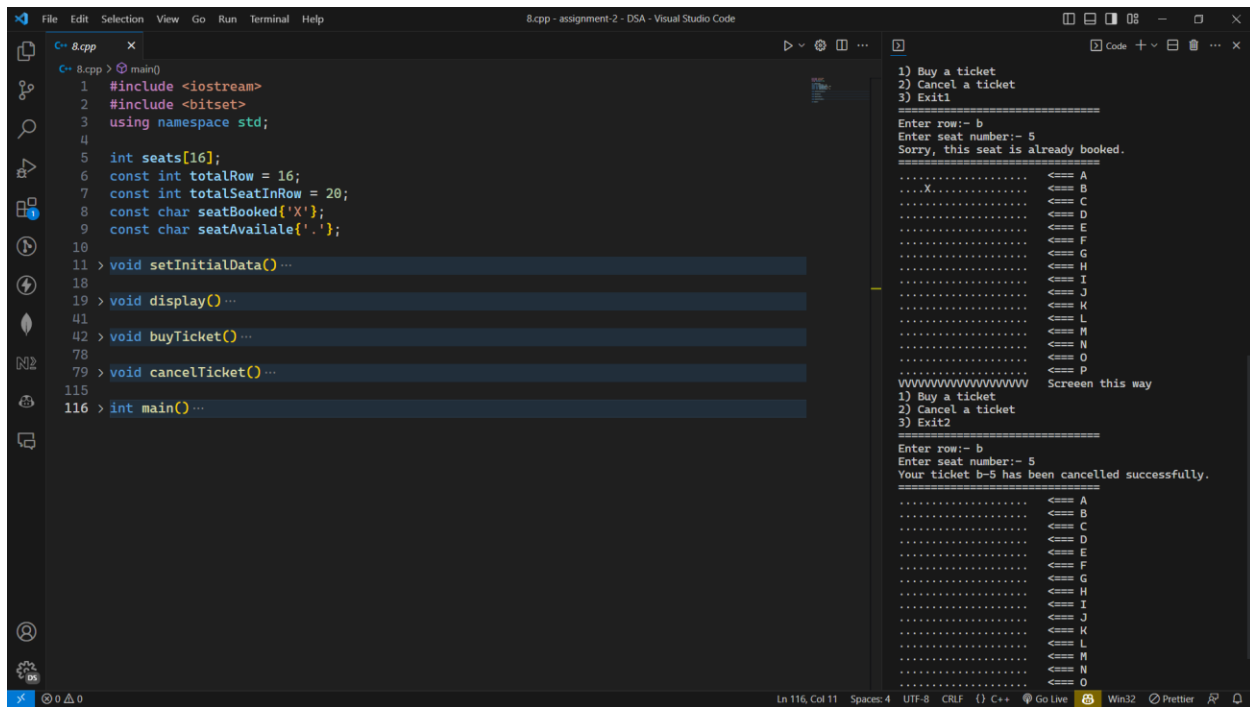
```
8.cpp
1 #include <iostream>
2 #include <bitset>
3 using namespace std;
4
5 int seats[16];
6 const int totalRow = 16;
7 const int totalSeatInRow = 20;
8 const char seatBooked{'X'};
9 const char seatAvailale{'.'};
10
11 > void setInitialData() ...
18
19 > void display() ...
41
42 > void buyTicket() ...
78
79 > void cancelTicket() ...
115
116 > int main() ...
```

Output:

```
PS C:\Users\kishan_HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2> cd "c:\Users\kishan_HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2" & if ($?) { g++ 8.cpp -o 8 } & if ($?) { .\8 }
..... <==== A
..... <==== B
..... <==== C
..... <==== D
..... <==== E
..... <==== F
..... <==== G
..... <==== H
..... <==== I
..... <==== J
..... <==== K
..... <==== L
..... <==== M
..... <==== N
..... <==== O
..... <==== P
.....
vvvvvvvvvvvvvvvvvvvv Screen this way
1) Buy a ticket
2) Cancel a ticket
3) Exit
Enter row:- b
Enter seat number:- 5
b-5 has been booked successfully.
.....
..... <==== A
..... <==== B
..... <==== C
..... <==== D
..... <==== E
..... <==== F
..... <==== G
..... <==== H
..... <==== I
..... <==== J
..... <==== K
..... <==== L
..... <==== M
..... <==== N
..... <==== O
..... <==== P
.....
vvvvvvvvvvvvvvvvvvvv Screen this way
1) Buy a ticket
2) Cancel a ticket
```

Name : Kishan Vaghamashi

Student ID: 202312014



```
File Edit Selection View Go Run Terminal Help
8.cpp - assignment-2 - DSA - Visual Studio Code

C++ 8.cpp x
1 #include <iostream>
2 #include <bitset>
3 using namespace std;
4
5 int seats[16];
6 const int totalRow = 16;
7 const int totalSeatInRow = 20;
8 const char seatBooked{'X'};
9 const char seatAvailale{'.'};
10
11 > void setInitialData()...
18
19 > void display()...
41
42 > void buyTicket()...
78
79 > void cancelTicket()...
115
116 > int main()...

1) Buy a ticket
2) Cancel a ticket
3) Exit1
=====
Enter row:- b
Enter seat number:- 5
Sorry, this seat is already booked.
=====
..... <=== A
...X..... <=== B
..... <=== C
..... <=== D
..... <=== E
..... <=== F
..... <=== G
..... <=== H
..... <=== I
..... <=== J
..... <=== K
..... <=== L
..... <=== M
..... <=== N
..... <=== O
..... <=== P
WWWWWWWWWWWW Screen this way
1) Buy a ticket
2) Cancel a ticket
3) Exit2
=====
Enter row:- b
Enter seat number:- 5
Your ticket b-5 has been cancelled successfully.
=====
..... <=== A
..... <=== B
..... <=== C
..... <=== D
..... <=== E
..... <=== F
..... <=== G
..... <=== H
..... <=== I
..... <=== J
..... <=== K
..... <=== L
..... <=== M
..... <=== N
..... <=== O
..... <=== P
=====
1) Buy a ticket
2) Cancel a ticket
3) Exit1
=====
```

9)

```
#include <bits/stdc++.h>
using namespace std;

vector<int> pendingOrders;

void placeOrder()
{
    int orderNumber = 1 + pendingOrders.size();
    pendingOrders.push_back(orderNumber);
    cout << "=====\n";
    cout << "Your order is placed. Order Number:- " << orderNumber << "\n";
    cout << "=====\n";
    cin.ignore();
}

void displayPendingOrders()
{
    cout << "=====\n";
    cout << "Pending orders:- "
        << "\n";
    for (int orderNumber : pendingOrders)
    {
        cout << "> Order Number: " << orderNumber << "\n";
    }
    cout << "Total pending orders:- " << pendingOrders.size() << "\n";
}
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
    cout << "=====\n";
}

void serveOrder()
{
    cout << "=====\n";
    cout << "Pending orders:\n";
    for (int orderNumber : pendingOrders)
    {
        cout << "=> Order Number: " << orderNumber << "\n";
    }

    cout << "\nPick order: ";
    int pickedOrder;
    cin >> pickedOrder;

    auto it = find(pendingOrders.begin(), pendingOrders.end(), pickedOrder);
    if (it != pendingOrders.end())
    {
        pendingOrders.erase(it);
        cout << "Order " << pickedOrder << " has been served.\n";
    }
    else
    {
        cout << "Invalid order number.\n";
    }
    cin.ignore();
    cout << "=====\n";
}

int main()
{
    int choice{0};
    bool exit = false;
    while (1)
    {
        if (exit)
            break;

        cout << "1) Place order\n2) Pending orders\n3) Serve order\n4) Exit\nEnter
your pick: ";
        cin >> choice;

        switch (choice)
        {
            case 1:
                placeOrder();
                break;

            case 2:
```

Name : Kishan Vaghamashi

Student ID: 202312014

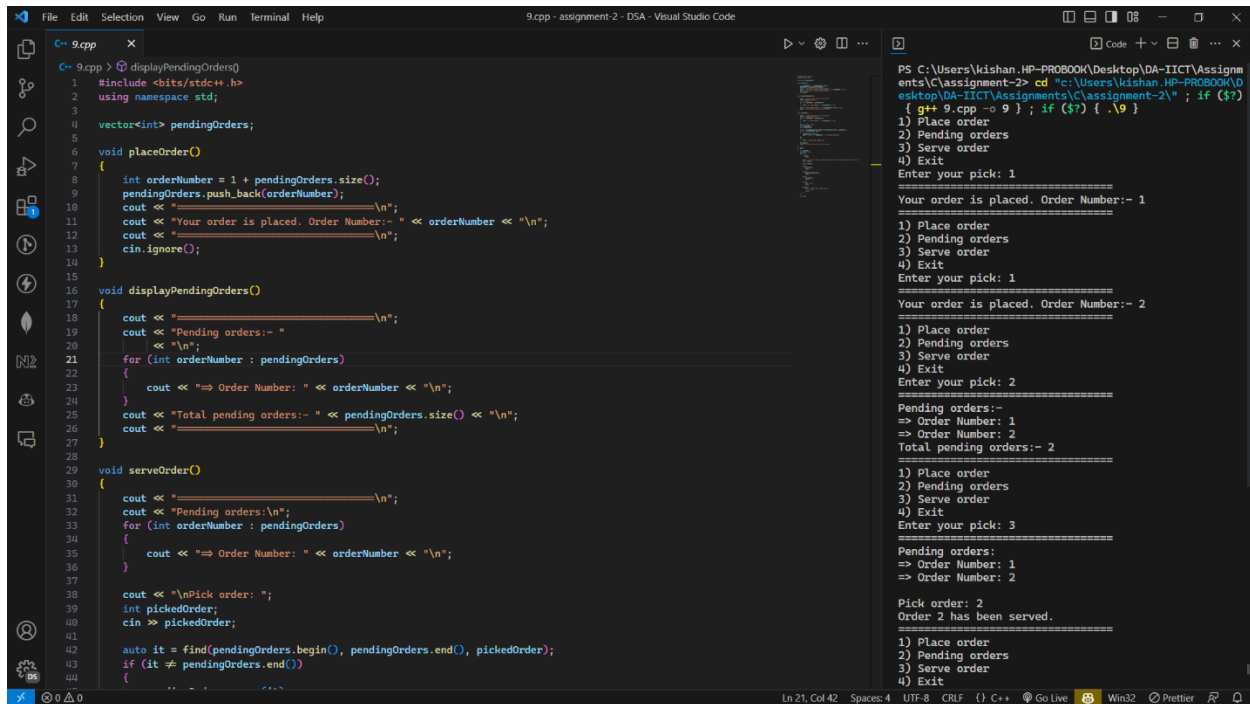
```
        displayPendingOrders();
        break;

    case 3:
        serveOrder();
        break;

    case 4:
        exit = true;
        break;

    default:
        cout << "Please enter valid choice."
              << "\n";
        break;
    }
}

return 0;
}
```



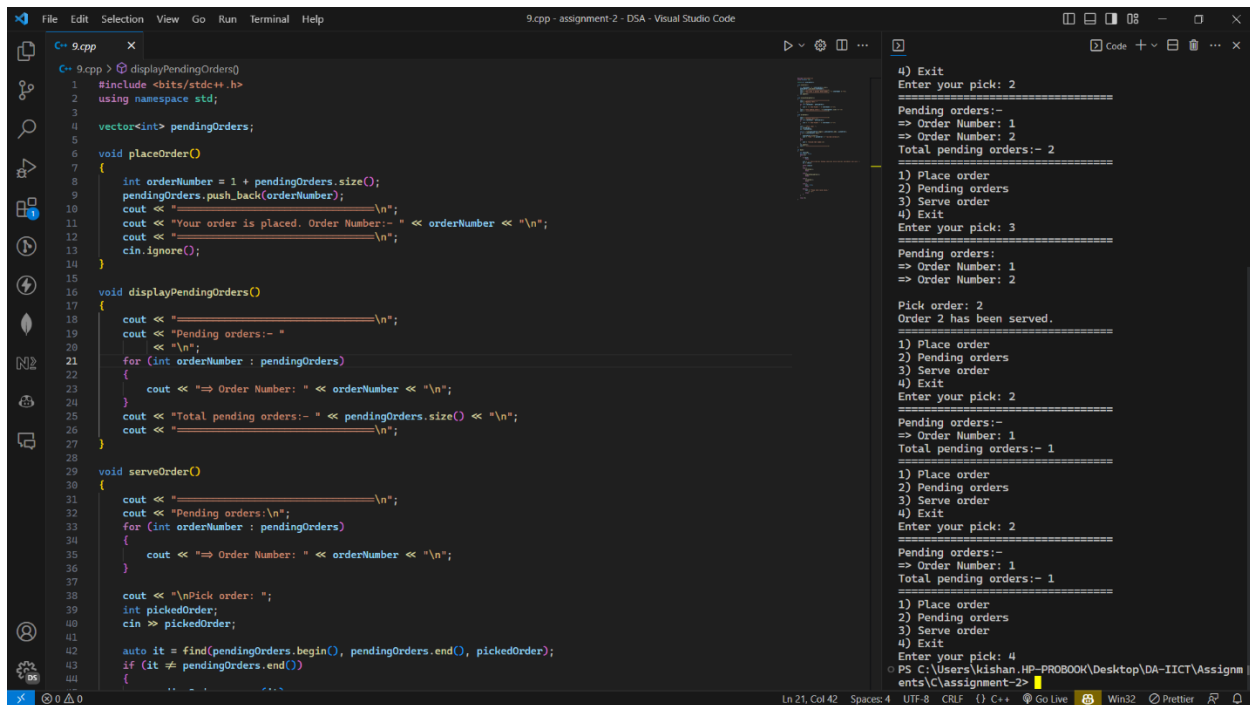
```
File Edit Selection View Go Run Terminal Help
9.cpp - assignment-2 - DSA - Visual Studio Code

C++ 9.cpp
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 vector<int> pendingOrders;
5
6 void placeOrder()
7 {
8     int orderNumber = 1 + pendingOrders.size();
9     pendingOrders.push_back(orderNumber);
10    cout << "\n";
11    cout << "Your order is placed. Order Number:- " << orderNumber << "\n";
12    cout << "\n";
13    cin.ignore();
14 }
15
16 void displayPendingOrders()
17 {
18     cout << "\n";
19     cout << "Pending orders:- "
20          << "\n";
21     for (int orderNumber : pendingOrders)
22     {
23         cout << "=> Order Number: " << orderNumber << "\n";
24     }
25     cout << "Total pending orders:- " << pendingOrders.size() << "\n";
26     cout << "\n";
27 }
28
29 void serveOrder()
30 {
31     cout << "\n";
32     cout << "Pending orders:\n";
33     for (int orderNumber : pendingOrders)
34     {
35         cout << "=> Order Number: " << orderNumber << "\n";
36     }
37
38     cout << "\nPick order: ";
39     int pickedOrder;
40     cin >> pickedOrder;
41
42     auto it = find(pendingOrders.begin(), pendingOrders.end(), pickedOrder);
43     if (it != pendingOrders.end())
44     {
45         // ...
46     }
47 }
```

```
PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2> cd "C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignments\assignment-2\" ; if ($?) { g++ 9.cpp -o 9 } ; if ($?) { 9 }
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 1
=====
Your order is placed. Order Number:- 1
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 1
=====
Your order is placed. Order Number:- 2
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 2
=====
Pending orders:-
=> Order Number: 1
=> Order Number: 2
Total pending orders:- 2
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 3
=====
Pending orders:
=> Order Number: 1
=> Order Number: 2
Pick order: 2
Order 2 has been served.
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
```

Name : Kishan Vaghamashi

Student ID: 202312014



```
9.cpp - assignment-2 - DSA - Visual Studio Code
C++ 9.cpp
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 vector<int> pendingOrders;
5
6 void placeOrder()
7 {
8     int orderNumber = 1 + pendingOrders.size();
9     pendingOrders.push_back(orderNumber);
10    cout << "=====\n";
11    cout << "Your order is placed. Order Number:- " << orderNumber << "\n";
12    cout << "=====\n";
13    cin.ignore();
14 }
15
16 void displayPendingOrders()
17 {
18    cout << "=====\n";
19    cout << "Pending orders:- "
20         << "\n";
21    for (int orderNumber : pendingOrders)
22    {
23        cout << "=> Order Number: " << orderNumber << "\n";
24    }
25    cout << "Total pending orders:- " << pendingOrders.size() << "\n";
26    cout << "=====\n";
27 }
28
29 void serveOrder()
30 {
31    cout << "=====\n";
32    cout << "Pending orders:\n";
33    for (int orderNumber : pendingOrders)
34    {
35        cout << "=> Order Number: " << orderNumber << "\n";
36    }
37
38    cout << "\nPick order: ";
39    int pickedOrder;
40    cin >> pickedOrder;
41
42    auto it = find(pendingOrders.begin(), pendingOrders.end(), pickedOrder);
43    if (it != pendingOrders.end())
44    {
45        pendingOrders.erase(it);
46    }
47 }
48
49 int main()
50 {
51    int row, col;
52    cin >> row >> col;
53    char arr[MAX][MAX] = {0};
54    bool flg = 0;
55    for (int i = 0; i < row; i++)
56    {
57        for (int j = 0; j < col; j++)
58        {
59            arr[i][j] = '*';
60        }
61    }
62    int round = 0;
63    row--, col--;
64    while (1)
65    {
66        int col_limit = col - (round + 1), row_limit = row - (round + 1);
67        int x1 = round, y1 = round, x2 = round, y2 = col - round, x3 = row - round,
68            y3 = round, x4 = row - round, y4 = col - round;
69        // ... (rest of the code)
70    }
71 }
```

4) Exit
Enter your pick: 2
=====
Pending orders:-
=> Order Number: 1
=> Order Number: 2
Total pending orders:- 2
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 3
=====
Pending orders:-
=> Order Number: 1
=> Order Number: 2
=====
Pick order: 2
Order 2 has been served.
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 2
=====
Pending orders:-
=> Order Number: 1
Total pending orders:- 1
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 2
=====
Pending orders:-
=> Order Number: 1
Total pending orders:- 1
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 2
=====
Pending orders:-
=> Order Number: 1
Total pending orders:- 1
=====
1) Place order
2) Pending orders
3) Serve order
4) Exit
Enter your pick: 4
PS C:\Users\kishan.HP-PROBOOK\Desktop\DA-IICT\Assignm
ents\C\assignment-2>

10)

```
#include <bits/stdc++.h>
using namespace std;
#define MAX 1000

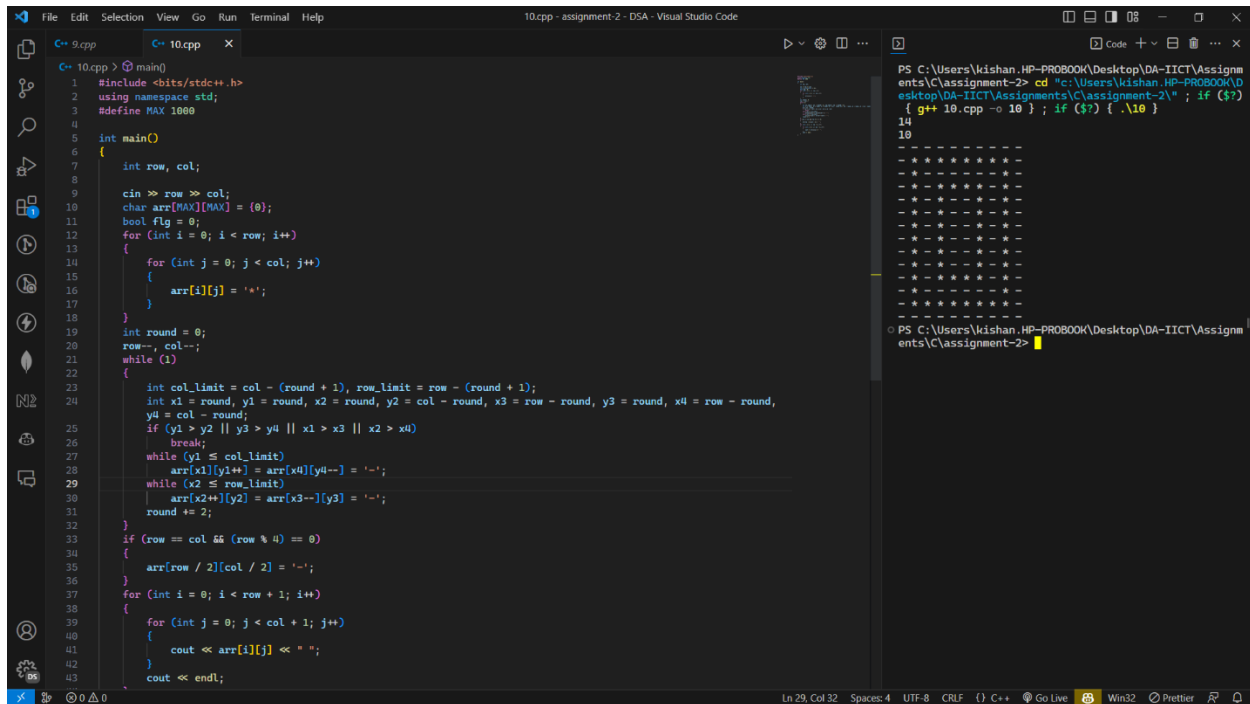
int main()
{
    int row, col;

    cin >> row >> col;
    char arr[MAX][MAX] = {0};
    bool flg = 0;
    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            arr[i][j] = '*';
        }
    }
    int round = 0;
    row--, col--;
    while (1)
    {
        int col_limit = col - (round + 1), row_limit = row - (round + 1);
        int x1 = round, y1 = round, x2 = round, y2 = col - round, x3 = row - round,
            y3 = round, x4 = row - round, y4 = col - round;
```

Name : Kishan Vaghamashi

Student ID: 202312014

```
        if (y1 > y2 || y3 > y4 || x1 > x3 || x2 > x4)
            break;
        while (y1 <= col_limit)
            arr[x1][y1++] = arr[x4][y4--] = '-';
        while (x2 <= row_limit)
            arr[x2++][y2] = arr[x3--][y3] = '-';
        round += 2;
    }
    if (row == col && (row % 4) == 0)
    {
        arr[row / 2][col / 2] = '-';
    }
    for (int i = 0; i < row + 1; i++)
    {
        for (int j = 0; j < col + 1; j++)
        {
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
}
```



The screenshot shows the Visual Studio Code editor with a C++ file named 10.cpp. The code is a recursive function to print a diamond pattern of asterisks. The output window shows the pattern for row 14, which is a diamond of 15 asterisks. The code is as follows:

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 #define MAX 1000
4
5 int main()
6 {
7     int row, col;
8
9     cin >> row >> col;
10    char arr[MAX][MAX] = {0};
11    bool flg = 0;
12    for (int i = 0; i < row; i++)
13    {
14        for (int j = 0; j < col; j++)
15        {
16            arr[i][j] = '*';
17        }
18    }
19    int round = 0;
20    row--, col--;
21    while (1)
22    {
23        int col_limit = col - (round + 1), row_limit = row - (round + 1);
24        int x1 = round, y1 = round, x2 = round, y2 = col - round, x3 = row - round, y3 = round, x4 = row - round,
25        y4 = col - round;
26        if (y1 > y2 || y3 > y4 || x1 > x3 || x2 > x4)
27            break;
28        while (y1 <= col_limit)
29            arr[x1][y1++] = arr[x4][y4--] = '-';
30        while (x2 <= row_limit)
31            arr[x2++][y2] = arr[x3--][y3] = '-';
32        round += 2;
33    }
34    if (row == col && (row % 4) == 0)
35    {
36        arr[row / 2][col / 2] = '-';
37    }
38    for (int i = 0; i < row + 1; i++)
39    {
40        for (int j = 0; j < col + 1; j++)
41        {
42            cout << arr[i][j] << " ";
43        }
44        cout << endl;
45    }
46 }
```

The output window shows the pattern for row 14, which is a diamond of 15 asterisks.