```
// Write a program in C++ to Create a student database of SE IT class (at least 10 records)
and perform the Binary search
// Using C++


#include <iostream>
#include <string>
using namespace std;

typedef struct student
{
    int roll_num;
    string name;
    float marks;
} stud;

void create(stud s[20], int n);
void display(stud s[20], int n);
void binary_search(stud s[20], int n, int key);

int main()
{
    stud s[20];
    int ch, n, key;

    do
    {
        cout << "\n 1) Create Student Database";
        cout << "\n 2) Display Student Records";
        cout << "\n 3) Binary Search by Roll Number";
        cout << "\n 4) Exit";
        cout << "\n Enter Your Choice: ";
        cin >> ch;

        switch (ch)
        {
        case 1:
            cout << "\n Enter the Number of Records (at least 10): ";
            cin >> n;
            if (n < 10)
            {
                cout << "Please enter at least 10 records.\n";
                break;
            }
            create(s, n);
            break;

        case 2:
            display(s, n);
            break;

        case 3:
            cout << "\n Enter the Roll Number to Search: ";
            cin >> key;
            binary_search(s, n, key);
            break;

        case 4:
            return 0;

        default:
            cout << "\n Invalid choice" << endl;
        }
    } while (ch != 4);
}

void create(stud s[20], int n)
{
    for (int i = 0; i < n; i++)
    {
        cout << "\n Enter the roll number: ";
        cin >> s[i].roll_num;
        cin.ignore();
        cout << " Enter the Name: ";
        getline(cin, s[i].name);
        cout << " Enter the marks: ";
        cin >> s[i].marks;
    }
}

void display(stud s[20], int n)
{
    cout << "\n\tRoll No\t\tName\t\tMarks";
    for (int i = 0; i < n; i++)
    {
        cout << "\n\t" << s[i].roll_num << "\t\t" << s[i].name << "\t\t" << s[i].marks;
    }
    cout << endl;
}

void binary_search(stud s[20], int n, int key)
{
    // Binary search requires sorted data — sort by roll number first (simple logic inside)
    stud temp;
    for (int i = 0; i < n - 1; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            if (s[i].roll_num > s[j].roll_num)
            {
                temp = s[i];
                s[i] = s[j];
                s[j] = temp;
            }
        }
    }

    int low = 0, high = n - 1, mid;
    bool found = false;

    while (low <= high)
    {
        mid = (low + high) / 2;
        if (s[mid].roll_num == key)
        {
            cout << "\nRecord Found!";
            cout << "\n\tRoll No\t\tName\t\tMarks";
            cout << "\n\t" << s[mid].roll_num << "\t\t" << s[mid].name << "\t\t" << s[mid].marks <<
endl;
            found = true;
            break;
        }
        else if (key < s[mid].roll_num)
            high = mid - 1;
        else
            low = mid + 1;
    }

    if (!found)
        cout << "\nRecord with Roll Number " << key << " not found.\n";
}
```