

1. Write a C++ program to create an employee database using structures concept.

```
#include <iostream>

#include <string>

using namespace std;

struct employee {

    string name, address;

    int id;

};

int main() {

    int n, c;

    cout << "\nEnter number of employees: ";

    cin >> n;

    cin.ignore();

    struct employee e[n];

    cout << "\nEnter Employee Data\n";

    for (int i = 0; i < n; i++) {

        cout << "\nEmployee " << i + 1;

        cout << "\n\tEnter Employee name: ";

        getline(cin, e[i].name);

        cout << "\n\tEnter Employee-Id: ";

        cin >> e[i].id;

        cin.ignore();

        cout << "\n\tEnter Employee Address: ";

        getline(cin, e[i].address);

    }

    int flag, se;
```

```

string add;

do {

    flag = 0;

    cout << "\nOperations Available\n1. Search By Employee Id\n2. Search By Employee
Address\n3. Exit\nEnter: ";

    cin >> c;

    cin.ignore();

    switch(c) {

        case 1:

            cout << "\nEnter Employee-Id: ";

            cin >> se;

            for (int i = 0; i < n; i++) {

                if (e[i].id == se) {

                    cout << "\nEmployee Name: " << e[i].name

                        << "\nEmployee-id: " << e[i].id

                        << "\nEmployee Address: " << e[i].address << endl;

                    flag = 1;

                    break;

                }

            }

            if (!flag) {

                cout << "\nEmployee Not Found\n";

            }

            break;

        case 2:

            cout << "\nEnter Employee Address: ";

```

```

        getline(cin, add);

        for (int i = 0; i < n; i++) {

            if (e[i].address == add) {

                cout << "\nEmployee Name: " << e[i].name

                    << "\nEmployee-id: " << e[i].id

                    << "\nEmployee Address: " << e[i].address << endl;

                flag = 1;

                break;

            }

        }

        if (!flag) {

            cout << "\nEmployee Not Found\n";

        }

        break;

    case 3:

        cout << "\nExiting the program\n";

        exit(0);

    default:

        cout << "\nInvalid choice\n";

    }

} while (1);

return 0;

}

```

```
"C:\Users\Justin D'souza\Desktop" X + v

Enter number of employees: 3
Enter Employee Data
Employee 1
    Enter Employee name: Rajeev Singh
    Enter Employee-Id: 4124
    Enter Employee Address: Panjim-Goa, India
Employee 2
    Enter Employee name: John Dias
    Enter Employee-Id: 4155
    Enter Employee Address: Mumbai, India
Employee 3
    Enter Employee name: Satish Gaonkar
    Enter Employee-Id: 4114
    Enter Employee Address: Margao-Goa, India

Operations Available
1. Search By Employee Id
2. Search By Employee Address
3. Exit
Enter: 1

Enter Employee-Id: 4114

Employee Name: Satish Gaonkar
Employee-id: 4114
Employee Address: Margao-Goa, India

Operations Available
1. Search By Employee Id
2. Search By Employee Address
3. Exit
Enter: 2

Enter Employee Address: Margao-Goa, India

Employee Name: Satish Gaonkar
Employee-id: 4114
Employee Address: Margao-Goa, India

Operations Available
1. Search By Employee Id
2. Search By Employee Address
3. Exit
Enter: 3

Exiting the program

Process returned 0 (0x0)   execution time : 191.726 s
Press any key to continue.
```

2. Write a Program to read names of users and units consumed and print out the charges with the names

```
#include <iostream>
```

```
#include <iomanip>
```

```

using namespace std;

struct user {

    string name;

    unsigned int units;

};

float calc(int info);

int main() {

    int n;

    cout << "\nEnter Number of users: ";

    cin >> n;

    cin.ignore();

    user u[n];

    cout << "\nEnter Details of users";

    for (int i = 0; i < n; i++) {

        cout << "\n\tEnter User's name: ";

        getline(cin, u[i].name);

        cout << "\n\tEnter electricity units consumed: ";

        cin >> u[i].units;

        cin.ignore();

    }

    cout << setw(20) << "Name" << setw(15) << "Bill(in Rs.)" << endl;

    for (int i = 0; i < n; i++) {

        cout << setw(20) << u[i].name << setw(15) << setprecision(3) << calc(u[i].units) << endl;

    }

    return 0;

}

```

```

float calc(int info) {

    float bill;

    if (info <= 100)

        bill = info * 60 / 100;

    else if (info <= 300)

        bill = ((100 * 60) + ((info - 100) * 80)) / 100;

    else

        bill = ((100 * 60) + (200 * 80) + ((info - 300) * 90)) / 100;

    if (bill < 50)

        bill = 50;

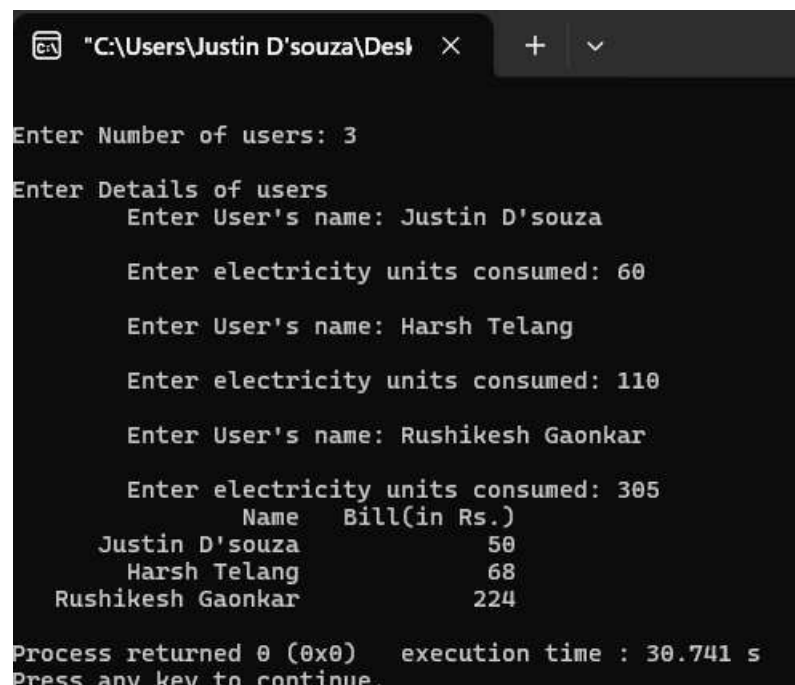
    else if (bill > 300)

        bill += (bill * 0.15);

    return bill;

}

```



The screenshot shows a Windows command prompt window with the following text:

```

C:\Users\Justin D'souza\Desktop>
Enter Number of users: 3
Enter Details of users
Enter User's name: Justin D'souza
Enter electricity units consumed: 60
Enter User's name: Harsh Telang
Enter electricity units consumed: 110
Enter User's name: Rushikesh Gaonkar
Enter electricity units consumed: 305
Name    Bill(in Rs.)
Justin D'souza    50
Harsh Telang      68
Rushikesh Gaonkar 224
Process returned 0 (0x0)   execution time : 30.741 s
Press any key to continue.

```

3. Write a C++ program to print a Fibonacci triangle

```
#include <iostream>
```

```

using namespace std;

int main() {

    int r;

    cout << "\nEnter number of rows: ";

    cin >> r;

    if (r >= 1) {

        for (int i=1; i <= r;i++){

            int a=1, b = 1, temp;

            cout << endl<< a << " ";

            if (i > 1){

                cout<<b<< " ";

            }

            for (int j=3; j<=i;j++) {

                temp=a+b;

                a=b;

                b=temp;

                cout<<temp<<" ";

            }

        }

    }

    else{

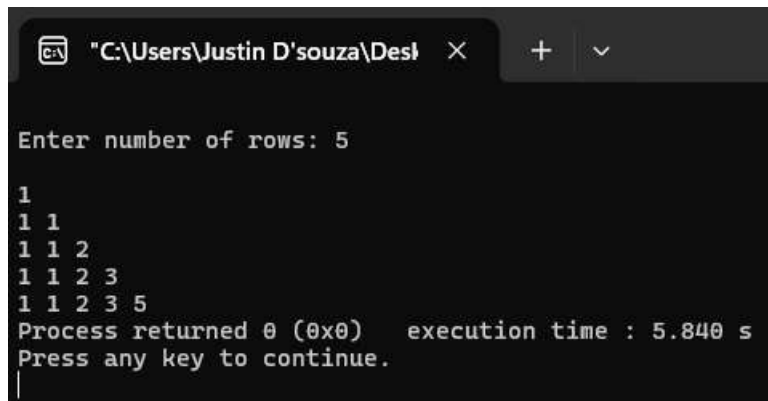
        cout<<"\nRows should be atleast greater than or equal to 1";

    }

    return 0;

}

```



```
"C:\Users\Justin D'souza\Desktop" X + v
Enter number of rows: 5
1
1 1
1 1 2
1 1 2 3
1 1 2 3 5
Process returned 0 (0x0) execution time : 5.840 s
Press any key to continue.
```

4. Write a C++ Program to interchange diagonals of a matrix (2D Array)

```
#include <iostream>

using namespace std;

int main(){

    int n;

    cout << "Enter the size of the matrix (n x n): ";

    cin >> n;

    int m[n][n],temp;

    cout << "Enter the elements of the matrix:" << endl;

    for (int i = 0; i < n; ++i) {

        for (int j = 0; j < n; ++j) {

            cin >> m[i][j];

        }

    }

    cout << "Original matrix:" << endl;

    for (int i = 0; i < n; ++i) {

        for (int j = 0; j < n; ++j) {

            cout<<m[i][j]<<"\t";

        }

        cout<<endl;

    }
```



```

    }

    cout<<"\nMatrix After Diagonals Interchanged:\n";

    for(int i=0;i<n;i++){

        temp=m[i][i];

        m[i][i]=m[i][n-i-1];

        m[i][n-i-1]=temp;

    }

    for (int i = 0; i < n; ++i) {

        for (int j = 0; j < n; ++j) {

            cout<<m[i][j]<<"\t";

        }

        cout<<endl;

    }

    return 0;

}

```

The screenshot shows a Windows command prompt window with the following text:

```

C:\Users\Justin D'souza\Desktop >
Enter the size of the matrix (n x n): 4
Enter the elements of the matrix:
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
Original matrix:
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
Matrix After Diagonals Interchanged:
0 0 0 1
0 0 1 0
0 1 0 0
1 0 0 0
Process returned 0 (0x0)   execution time : 26.229 s
Press any key to continue.

```