Name: Syed Abdul Muiz

ID: F24CSC001/28770

PROGRAM 01:

```cpp
#include <iostream>

using namespace std;

int main()

{

   int num = 5;


   // Use increment operator to display numbers from 5 to 10 vertically

   while (num <= 10)

       {

     cout << num << endl; // Output the current value of num on a new line

     num++;  // Add num by 1

   }


   return 0;

}
```

PROGRAM 02:

```cpp
#include <iostream>

#include <iomanip>

using namespace std;

int main()

{
```

```cpp
    double distance_km;

    // Input the distance in kilometers
    cout << "Enter the distance from home to SHU (in kilometers): ";
    cin >> distance_km;

    // Convert the distance into various units
    double distance_meters = distance_km * 1000;
    double distance_centimeters = distance_meters * 100;
    double distance_millimeters = distance_centimeters * 10;
    double distance_micrometers = distance_millimeters * 1000;

    // Output the distances in different units
    cout << fixed << setprecision(3);

    cout << "\nDistance in kilometers: " << distance_km << " km" << endl;
    cout << "Distance in meters: " << distance_meters << " m" << endl;
    cout << "Distance in centimeters: " << distance_centimeters << " cm" << endl;
    cout << "Distance in millimeters: " << distance_millimeters << " mm" << endl;
    cout << "Distance in micrometers: " << distance_micrometers << " µm" << endl;

    return 0;
}
```

PROGRAM 03:

```cpp
#include<iostream>
```

```cpp
using namespace std;

int main()

{

    const int numCourses = 5;  // Number of courses

    double creditHours[numCourses];  // Array to store credit hours of courses

    double gradePoints[numCourses];  // Array to store grade points earned in each course

    double totalCredits = 0.0, weightedSum = 0.0, sgpa = 0.0;


    // Input the credit hours and grade points for each course

    cout << "Enter the Credit Hours and Grade Points for each of the 5 courses:\n";

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

        cout << "Course: " << i + 1 << "\n";

        cout << "Credit Hours: ";

        cin >> creditHours[i];

        cout << "Grade Points: ";

        cin >> gradePoints[i];


        // Calculate the weighted sum and total credits

        weightedSum += creditHours[i] * gradePoints[i];

        totalCredits += creditHours[i];

    }


    // Calculate SGPA

    if (totalCredits > 0)

            {
```

```cpp
        sgpa = weightedSum / totalCredits;

        cout << "\nThe SGPA for the semester is: " << sgpa << endl;

    }

        else

        {

        cout << "Error: Total credits cannot be zero!" << endl;

    }


    return 0;

}
```

PROGRAM 06:

```cpp
#include <iostream>

using namespace std;

int main()

{

    int num1, num2, num3;


    cout << "Enter the first number: ";

    cin >> num1;


    cout << "Enter the second number: ";

    cin >> num2;


    cout << "Enter the third number: ";

    cin >> num3;
```

```cpp
    if (num1 == num2)

        {

    if (num2 == num3)

                {

        cout << "All values are equal." << endl;

    }

                else

                {

        cout << "They all are not equal." << endl;

    }

  }

  else

  {

        cout<<"They all are not equal."<<endl;

        }


    return 0;

}
```

PROGRAM 07:

```cpp
#include<iostream>

using namespace std;

int main()

{

        char m;
```

```cpp
        cout<<"Enter m for Male: ";

        cin>>m;


        if(m == 'm')

        {

                cout<<"You are Male."<<endl;

        }


        else

        {

                cout<<"You are Female."<<endl;

        }


        return 0;

}
```

PROGRAM 08:

```cpp
#include <iostream>

using namespace std;

int main()

{

    int marks;


    cout << "Enter marks obtained by the student (out of 100): ";

    cin >> marks;
```

```cpp
if (marks >= 0 && marks <= 100)

    {

  if (marks >= 90)

            {

    cout << "Grade: A+" << endl;

  }

  else if (marks >= 70)

            {

    cout << "Grade: A" << endl;

  }

  else if (marks >= 50)

            {

    cout << "Grade: B" << endl;

  }

  else

            {

    cout << "Grade: F" << endl;

  }

}

      else

      {

  cout << "Invalid marks entered. Marks should be between in 0 and 100." << endl;

}
```

```
    return 0;

}



PROGRAM 09:

#include <iostream>

using namespace std;

int main()

{

    char gender, city;

    int age;

    double salary, netSalary;


    cout << "Enter the current salary: ";

    cin >> salary;


    cout << "Enter gender (F/M): ";

    cin >> gender;


    cout << "Enter age: ";

    cin >> age;


    cout << "Enter city (K for Karachi, H for Hyderabad, S for Sukker, G for Ghotki): ";

    cin >> city;


    if (gender == 'F' || gender == 'f')
```

```cpp
    {

if ((age >= 25 && age <= 35) && (city == 'K' || city == 'H' || city == 'k' || city == 'h'))

        {

    netSalary = salary + 2000;

    cout << "Net salary (after addition): " << netSalary << endl;

  }

            else

        {

    cout << "Present salary: " << salary << endl;

  }

}

else if (gender == 'M' || gender == 'm')

    {

if ((age >= 25 && age <= 40) && (city == 'S' || city == 'G' || city == 's' || city == 'g'))

        {

    netSalary = salary + 2500;

    cout << "Net salary (after addition): " << netSalary << endl;

  }

            else

        {

    cout << "Present salary: " << salary << endl;

  }

}

else

    {
```

```cpp
        cout << "Invalid gender input." << endl;

    }



    return 0;

}
```

PROGRAM 11:

```cpp
#include <iostream>

#include <string>

using namespace std;



int main() {

    string enteredID, enteredPassword;

    string validID = "user123";

    string validPassword = "pass123";

    string userName = "John Doe";



    cout << "Enter your ID: ";

    cin >> enteredID;



    switch (enteredID == validID) {

        case true:

            {

                cout << "Enter your password: ";

                cin >> enteredPassword;
```

```cpp
        switch (enteredPassword == validPassword) {

            case true:

                cout << "Welcome, " << userName << "!" << endl;

                break;

            case false:

                cout << "Incorrect Password!" << endl;

                break;

        }

    }

    break;



    case false:

    cout << "Incorrect ID!" << endl;

    break;

}


    return 0;

}
```

PROGRAM 13:

```cpp
#include <iostream>

using namespace std;

int main()

{

    int mainOption, settingsOption, displayOption;
```

```cpp
cout << "Welcome to the Game!" << endl;

cout << "Please select an option:" << endl;

cout << "1. Start Game" << endl;

cout << "2. Settings" << endl;

cout << "3. Exit" << endl;

cout << "Enter your choice (1-3): ";

cin >> mainOption;


switch (mainOption)

    {

  case 1:

     cout << "Starting the game..." << endl;

     break;


  case 2:

     cout << "Settings Menu:" << endl;

     cout << "1. Display" << endl;

     cout << "2. Sound" << endl;

     cout << "3. Back to Main Menu" << endl;

     cout << "Enter your choice (1-3): ";

     cin >> settingsOption;


     switch (settingsOption)

                {
```

```cpp
case 1:

    cout << "Display Settings:" << endl;

    cout << "1. Graphics" << endl;

    cout << "2. Resolution" << endl;

    cout << "3. Back to Settings Menu" << endl;

    cout << "Enter your choice (1-3): ";

    cin >> displayOption;


    switch (displayOption)

                                {

        case 1:

            cout << "Graphics Settings:" << endl;

            cout << "1. High" << endl;

            cout << "2. Medium" << endl;

            cout << "3. Low" << endl;

            cout << "Enter your choice (1-3): ";

            int graphicsChoice;

            cin >> graphicsChoice;

            switch (graphicsChoice)

                                        {

                case 1: cout << "Graphics set to High." << endl; break;

                case 2: cout << "Graphics set to Medium." << endl; break;

                case 3: cout << "Graphics set to Low." << endl; break;

                default: cout << "Invalid choice." << endl; break;

            }
```

```cpp
        break;


    case 2:

        cout << "Resolution Settings:" << endl;

        cout << "1. 1920x1080" << endl;

        cout << "2. 1280x720" << endl;

        cout << "3. 800x600" << endl;

        cout << "Enter your choice (1-3): ";

        int resolutionChoice;

        cin >> resolutionChoice;

        switch (resolutionChoice) {

            case 1: cout << "Resolution set to 1920x1080." << endl; break;

            case 2: cout << "Resolution set to 1280x720." << endl; break;

            case 3: cout << "Resolution set to 800x600." << endl; break;

            default: cout << "Invalid choice." << endl; break;

        }

        break;


    case 3:

        cout << "Returning to Settings Menu." << endl;

        break;


    default:

        cout << "Invalid choice." << endl;

        break;
```

```
                }

                break;


            case 2:

                cout << "Sound Settings (Placeholder for future options)." << endl;

                break;


            case 3:

                cout << "Returning to Main Menu." << endl;

                break;


            default:

                cout << "Invalid choice." << endl;

                break;

            }

            break;


        case 3:

            cout << "Exiting the game. Goodbye!" << endl;

            break;


        default:

            cout << "Invalid choice. Please enter a valid option (1-3)." << endl;

            break;

        }
```

```
    return 0;

}
```

PROGRAM 12:

```
#include <iostream>

#include <cmath>  // for square root

using namespace std;


int main() {

    int num, choice;


    cout << "Enter a number: ";

    cin >> num;


    cout << "\nChoose an option from the menu:\n";

    cout << "1. Calculate the square root\n";

    cout << "2. Calculate the cube\n";

    cout << "3. Print the number 50 times\n";

    cout << "4. Print the multiplication table\n";

    cout << "Enter your choice (1-4): ";

    cin >> choice;


    switch (choice) {

        case 1:

            // Calculate the square root
```

```cpp
        if (num >= 0) {

            cout << "The square root of " << num << " is " << sqrt(num) << endl;

        } else {

            cout << "Error: Cannot calculate the square root of a negative number." << endl;

        }

        break;


    case 2:

        // Calculate the cube

        cout << "The cube of " << num << " is " << num * num * num << endl;

        break;


    case 3:

        // Print the number 50 times

        cout << "Printing the number 50 times:" << endl;

        for (int i = 0; i < 50; ++i) {

            cout << num << " ";

        }

        cout << endl;

        break;


    case 4:

        // Print the multiplication table of the number

        cout << "Multiplication table for " << num << ":" << endl;

        for (int i = 1; i <= 10; ++i) {
```

```cpp
        cout << num << " * " << i << " = " << num * i << endl;

      }

      break;


    default:

      // Handle invalid choices

      cout << "Invalid choice." << endl;

      break;

  }


  return 0;

}
```

PROGRAM 05:

```cpp
#include <iostream>

using namespace std;

int main()

{

  double A, b, x, y, a, c, d;


  cout << "Enter the values for b, x, y, a, c, d: ";

  cin >> b >> x >> y >> a >> c >> d;


  A = 7 * 7 * b * (x * y + a) / c - 0.8 + 2 * b * d * (x + a) * (1 / y);


  cout << "The value of A is: " << A << endl;
```

```
    return 0;

}
```

PROGRAM 04:

```
#include <iostream>

#include <cmath>  // For sqrt() function

using namespace std;


int main() {


    float a = 2.5, b = 5.0;

    int c;




    cout << "Enter the value of c: ";

    cin >> c;




    float discriminant = b * b - 4 * a * c;


    if (discriminant >= 0)
        {
        float sqrt_discriminant = sqrt(discriminant);

        float x1 = (-b + sqrt_discriminant) / (2 * a);  // First root (with +)

        float x2 = (-b - sqrt_discriminant) / (2 * a);  // Second root (with -)
```

```cpp
        // Output the two roots

        cout << "The roots of the quadratic equation are: " << endl;

        cout << "x1 = " << x1 << endl;

        cout << "x2 = " << x2 << endl;

    } else {

        // If the discriminant is negative, no real roots exist

        cout << "No real roots exist because the discriminant is negative." << endl;

    }


    return 0;

}
```