Experiment No 2

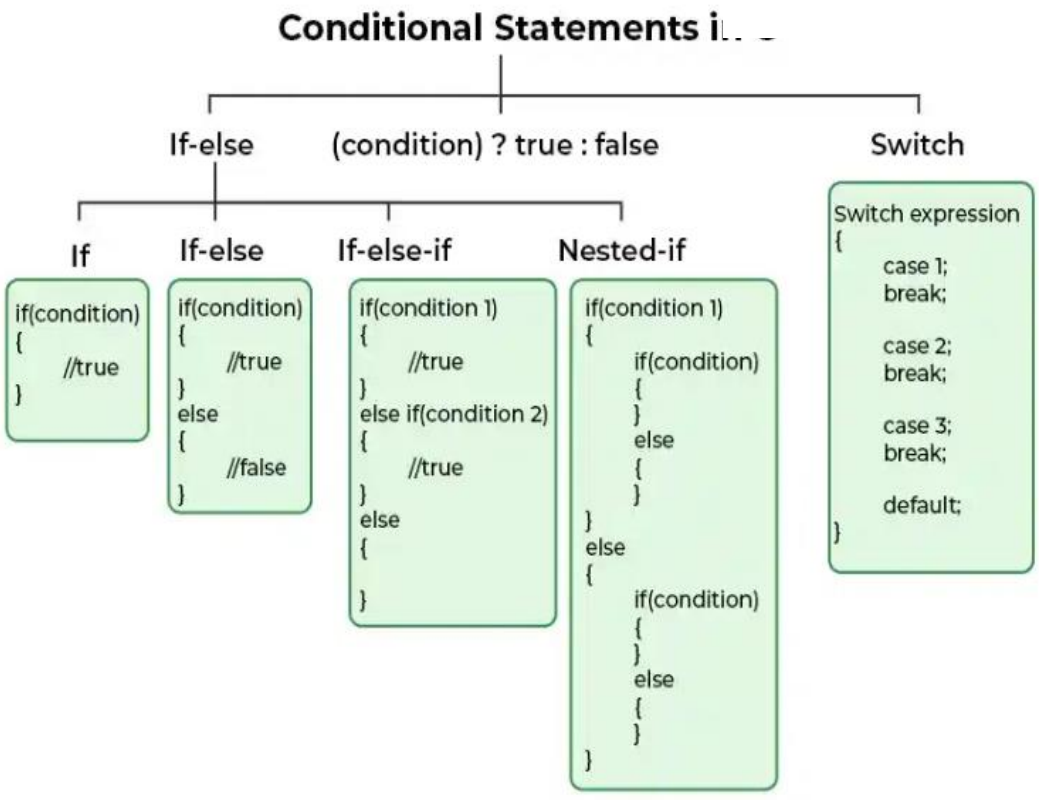
Aim: To implement the control statements in CPP.

Theory:

The **conditional statements** (also known as decision control structures) such as if, if else, switch, etc. are used for decision-making purposes in C++ programs.They are also known as Decision-Making Statements and are used to evaluate one or more conditions and make the decision whether to execute a set of statements or not. These decision-making statements in programming languages decide the direction of the flow of program execution.

Need of Conditional Statements

There come situations in real life when we need to make some decisions and based on these decisions, we decide what should we do next. Similar situations arise in programming also where we need to make some decisions and based on these decisions we will execute the next block of code. For example, in C if x occurs then execute y else execute z. There can also be multiple conditions like in C if x occurs then execute p, else if condition y occurs execute q, else execute r. This condition of C else-if is one of the many ways of importing multiple conditions.



if statement:

The if in C++ is a decision-making statement that is used to execute a block of code based on the value of the given expression. It is one of the core concepts of C++ programming and is used to include conditional code in our program.

Syntax of if Statement

if(condition)

```
{
   // if body
   // Statements to execute if condition is true
}
```

if-else statement:

The if-else statement is a decision-making statement that is used to decide whether the part of the code will be executed or not based on the **specified condition (test expression)**. If the given condition is true, then the code inside the if block is executed, otherwise the code inside the else block is executed.

Syntax of if-else

```
if (condition)
{
   // code executed when the condition is true
}
else
{
   // code executed when the condition is false
}
```

if else if ladder

The **if else if** ladder in C++ programming is used to test a series of conditions sequentially. Furthermore, if a condition is tested only when all previous if conditions in the if-else ladder are false. If any of the conditional expressions evaluate to be true, the appropriate code block will be executed, and the entire if-else ladder will be terminated.

Syntax of if else if ladder:

```
// any if-else ladder starts with an if statement only
if(condition)
{
     ……..
}
else if(condition)
{
 // this else if will be executed when condition in if is false and
 // the condition of this else if is true
}
.... // once if-else ladder can have multiple else if
else
{ // at the end we put else
    …….....
}
```

Switch Statement:

Switch case statement evaluates a given expression and based on the evaluated value(matching a certain condition), it executes the statements associated with it. Basically, it is used to perform different actions based on different conditions(cases).

- Switch case statements follow a selection-control mechanism and allow a value to change control of execution.
- They are a substitute for long if statements that compare a variable to several integral values.
- The switch statement is a multiway branch statement. It provides an easy way to dispatch execution to different parts of code based on the value of the expression.

Syntax of switch Statement

```
switch(expression)
{
        case value1: statement_1;
                break;
        case value2: statement_2;
                break;
        .
        .
        .
        case value_n: statement_n;
                break;
        default: default_statement;
}
```

Practical Related Questions:

1. Explain break statement with its syntax
2. Explain continue statements with its syntax.
3. Write a note on goto statement.
4. Draw a flow chart for switch case.

Programs :

1. Write a program that prompts the user for a number. If the number is even, print "The number is even. " Otherwise, print "The number is odd. **(if else statement)**
2. Design a program that simulates a simple grade calculator. Ask the user for their exam score (out of 100). Based on the score, print the corresponding letter grade:
   - A: 90-100
   - B: 80-89
   - C: 70-79
   - D: 60-69
   - F: Below 60 **(if-else ladder)**
3. Design a program that acts as a simple menu driven program to find square, square root, cube , perform the corresponding calculation and exit the program using **switch case.**

Conclusion :

Hence, we learnt to implement the conditional statements of CPP.

PRACTICAL 2

1. **Write a program that prompts the user for a number. If the number is even, print "The number is even. " Otherwise, print "The number is odd.**

```
#include <iostream.h>
int main()
{
        int num;
        cout << "Enter a number: ";
        cin >> num;
        if (num % 2 == 0)
        {
            cout << "The number is even." << endl;
        }
        else
        {
            cout << "The number is odd." << endl;
        }

        return 0;
}
```

2. **Design a program in cpp that simulates a simple grade calculator. Ask the user for their exam score (out of 100) of 5 subjects .calculate the total and averade Based on the average, print the corresponding letter grade:**

1. **A: 90-100**
2. **B: 80-89**
3. **C: 70-79**
4. **D: 60-69**
5. **F: Below 60**

```
#include <iostream.h>
int main()
    {
        int subject1, subject2, subject3, subject4, subject5;
        float total, average;
        cout << "Enter marks for 1st subjects (out of 100): " << endl;
        cin >>subject1;
        cout << "Enter marks for 2nd subjects (out of 100): " << endl;
        cin >>subject2;
        cout << "Enter marks for 3rd subjects (out of 100): " << endl;
        cin >>subject3;
        cout << "Enter marks for 4th subjects (out of 100): " << endl;
        cin >>subject4;
```

```
        cout << "Enter marks for 5th subjects (out of 100): " << endl;
        cin >>subject5;
        total = subject1 + subject2 + subject3 + subject4 + subject5;
        average = total / 5;
        cout << "Total marks: " << total << endl;
        cout << "Average marks: " << average << endl;
        if (average >= 90)
        {
                cout << "Grade: A" << endl;
        }
        else if (average >= 80)
        {
                cout << "Grade: B" << endl;
        }
        else if (average >= 70)
        {
                cout << "Grade: C" << endl;
        }
        else if (average >= 60)
        {
                cout << "Grade: D" << endl;
        }
        else
        {
                cout << "Grade: F" << endl;
        }
        return 0;
}
```

3. **Design a program that acts as a simple menu driven program to find square, square root, cube , perform the corresponding calculation and exit the program using switch case.**

```
#include <iostream.h>
#include <math.h>
int main()
{
        int choice, num;
        double result;
        do
        {
                cout << "\nMenu:\n";
                cout << "1. Find square\n";
                cout << "2. Find square root\n";
                cout << "3. Find cube\n";
```

```
            cout << "4. Exit\n";
            cout << "Enter your choice: ";
            cin >> choice;
            switch (choice)
            {
                case 1:
                    cout << "Enter a number: ";
                    cin >> num;
                    result = num * num;
                    cout << "Square of " << num << " is: " << result << endl;
                    break;
                case 2:
                    cout << "Enter a number: ";
                    cin >> num;
                    if (num >= 0)
                    {
                        result = sqrt(num);
                        cout << "Square root of " << num << " is: " << result << endl;
                    }
                    else
                    {
                        cout << "Square root of negative number is not defined." ;
                    }
                    break;
                case 3:
                    cout << "Enter a number: ";
                    cin >> num;
                    result = num * num * num;
                    cout << "Cube of " << num << " is: " << result << endl;
                    break;
                case 4:
                    cout << "Exiting program..." << endl;
                    break;
                default:
                    cout << "Invalid choice. Please try again." << endl;
            }
        } while (choice != 4);
            return 0;
}
```