

COMPUTER SCIENCE : C++

C++ Program to Make a Simple Calculator to Add, Subtract, Multiply or Divide Using switch...case

Example: Simple Calculator using switch statement

```
# include <iostream.h>
using namespace std;

void main()
{
    char op;
    float num1, num2;

    cout << "Enter operator either + or - or * or /: ";
    cin >> op;

    cout << "Enter two operands: ";
    cin >> num1 >> num2;

    switch(op)
    {
        case '+':
            cout << num1+num2;
            break;

        case '-':
            cout << num1-num2;
```

```

        break;

    case '*':
        cout << num1*num2;
        break;

    case '/':
        cout << num1/num2;
        break;

    default:
        // If the operator is other than +, -, * or /, error message is shown
        cout << "Error! operator is not correct";
        break;
}

return 0;
}

```

Output

```

Enter operator either + or - or * or divide : -
Enter two operands:
3.4
8.4
3.4 - 8.4 = -5.0

```

This program takes an operator and two operands from user.

The operator is stored in variable `op` and two operands are stored in `num1` and `num2` respectively.

Then, switch...case statement is used for checking the operator entered by user.

If user enters + then, statements for `case: '+'` is executed and program is terminated.

If user enters - then, statements for `case: '-'` is executed and program is terminated.

This program works similarly for * and / operator. But, if the operator doesn't matches any of the four character [+, -, * and /], default statement is executed which displays error message.

C++ Program to Convert Binary Number to Octal and vice-versa

Program to Convert Binary to Octal

In this program, we will first convert the binary number to decimal. Then, the decimal number is converted to octal.

```
#include <iostream.h>
#include <cmath.h>

using namespace std;

int convertBinarytoOctal(long long);
void main()
{
    long long binaryNumber;

    cout << "Enter a binary number: ";
    cin >> binaryNumber;

    cout << binaryNumber << " in binary = " << convertBinarytoOctal(binaryNumber)
    << " in octal ";

    return 0;
}

int convertBinarytoOctal(long long binaryNumber)
{
    int octalNumber = 0, decimalNumber = 0, i = 0;

    while(binaryNumber != 0)
    {
        decimalNumber += (binaryNumber%10) * pow(2,i);
        ++i;
        binaryNumber/=10;
    }
}
```

```
    i = 1;

    while (decimalNumber != 0)
    {
        octalNumber += (decimalNumber % 8) * i;
        decimalNumber /= 8;
        i *= 10;
    }

    return octalNumber;
}
```

Output

```
Enter a binary number: 10001
10001 in binary = 21 in octal
```

The binary number entered by the user is passed to `convertBinaryToOctal()` function. And, this function converts the number to octal and returns to the `main()` function

Program to Convert Octal to Binary

In this program, the octal number is converted to decimal at first. Then, the decimal number is converted to binary number.

```
#include <iostream.h>
#include <cmath.h>

using namespace std;

long long convertOctalToBinary(int);
void main()
{
    int octalNumber;

    cout << "Enter an octal number: ";
    cin >> octalNumber;

    cout << octalNumber << " in octal = " << convertOctalToBinary(octalNumber) <<
    "in binary";

    return 0;
}

long long convertOctalToBinary(int octalNumber)
{
    int decimalNumber = 0, i = 0;
    long long binaryNumber = 0;

    while(octalNumber != 0)
    {
        decimalNumber += (octalNumber%10) * pow(8,i);
        ++i;
        octalNumber/=10;
    }

    i = 1;

    while (decimalNumber != 0)
    {
        binaryNumber += (decimalNumber % 2) * i;
    }
}
```

```
        decimalNumber /= 2;
        i *= 10;
    }

    return binaryNumber;
}
```

Output

```
Enter an octal number: 54
54 in octal = 101100
```

The octal number entered by the user is passed to `convertOctalToBinary()` function. And, this function converts the number to binary and returns the `main()` function