Problem Statement

Given a decimal number as input, we need to write a program to convert the given decimal number into equivalent octal number. i.e convert the number with base value 10 to base value 8.

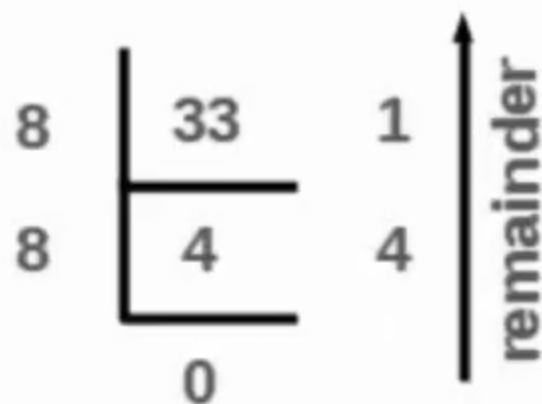# Problem Statement

Given a decimal number as input, we need to write a program to convert the given decimal number into equivalent octal number. i.e convert the number with base value 10 to base value 8.

The base value of a number system determines the number of digits used to represent a numeric value.

For example, the binary number system uses two digits 0 and 1, octal number system uses 8 digits from 0-7 and decimal number system uses 10 digits 0-9 to represent any numeric value.

Algorithm

Decimal Number: 33

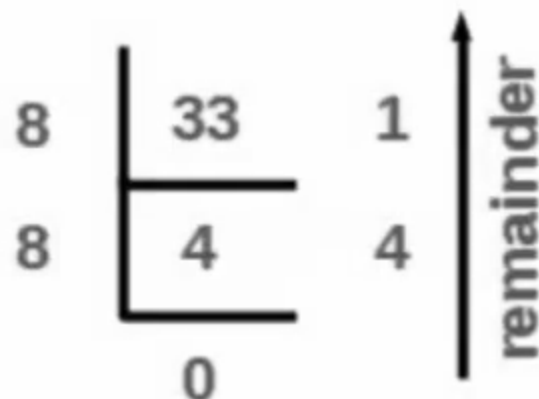| 8 | 33 | 1 |
| 8 | 4 | 4 |
| | 0 | |

remainder

Octal Number: 41

1. Store the remainder when the number is divided by 8 in an array.
2. Divide the number by 8.
3. Repeat the above two steps until the number is not equal to 0.
4. Print the array in reverse order now.

Google Slides

Algorithm

Decimal Number: 33

8	33	1
8	4	4
	0	

remainder

Octal Number: 41

1. Store the remainder when the number is divided by 8 in an array.
2. Divide the number by 8.
3. Repeat the above two steps until the number is not equal to 0.
4. Print the array in reverse order now.

Google Slides

Code

```cpp
// function to convert decimal to octal
void decToOctal(int n)
{

    // array to store octal number
    int octalNum[100];

    // counter for octal number array
    int i = 0;
    while (n != 0) {

        // storing remainder in octal array
        octalNum[i] = n % 8;
        n = n / 8;
        i++;
    }


    // printing octal number array in reverse order
    for (int j = i - 1; j >= 0; j--)
        cout << octalNum[j];

}
```

Code

```cpp
// function to convert decimal to octal
void decToOctal(int n)
{

    // array to store octal number
    int octalNum[100];

    // counter for octal number array
    int i = 0;
    while (n != 0) {

        // storing remainder in octal array
        octalNum[i] = n % 8;
        n = n / 8;
        i++;
    }

    // printing octal number array in reverse order
    for (int j = i - 1; j >= 0; j--)
        cout << octalNum[j];

}
```

Dry Run

octalNum[] = {}
n = 33
i = 0

Thank you for watching!
Please leave us your comments.