26-04-2024

OPERATORS :- It is a symbol which performs some operation on operands.

1. Arithmetic
2. Modulus / Modulo Division
3. Increment & Decrement
4. Relational
5. Logical
6. Bitwise
7. Conditional
8. Assignment
9. Special operators

1. ARITHMETIC :- + - * / these are binary operators.

We apply these operators on integers, floats & characters.

```
Ex :- main ()
      { int c;
        c = 'B' + 'A';
        printf ("%d", c);
      }
```

66
65

O/P :- 131

2. MODULUS ['%] :- It is a binary operator. It gives remainder

```
Ex :- main ()
      { int c;
        c = 5 % 2;
        printf ("%d", c);
      }
```

O/P :- 1

→ If Nr < Dr, Numerator is the remainder

```
Ex :- main ()
      { int c;
        c = 8 % 9;
        printf ("%d", c);
      }
```

O/P :-

->The sign of the remainder depends on only Numerator not on denominator

Ex:-

```
main()
{
printf("%d %d %d %d %d %d %d %d", 4%3, -4%3,
4%-3, -4%-3, 3%4, -3%4, 3%-4, -3%-4);
}
```

O/P:- 1 -1 1 -1 3 -3 3 -3

-> We should not apply % operator on float

Ex:- d = 5 % 2.5; -> It gives error

Ex:-

```
main()
{ int days, months, weeks;
  printf("Enter days:");
  scanf("%d", &days);
  months = days/30;
  days = days %30;
  weeks = days/7;
  days = days %7;
  printf("%d months %d weeks %d days",
          months, weeks, days);
}
```

Ex:- main ()

Enter min

o/p x hrs x min

```
{ int mins, hrs;
  printf ("enter no of minutes:");
  scanf ("%d", &min);
  hrs = min / 60;
  printf "%d min
  min = hrs * 60;
min hrs = min % 60;
  printf ("%d hours %d minutes", hrs, mins);
}
```

2
150
60

9
n = 2 5
m = 2

24

Q:- WAP to find sum of individual digits of 3 numbers:-

```
main ()
{ int n, m, s=0;
  printf ("enter n");
  scanf ("%d", &n);
  m = n % 10;
  s = s + m;
  n = n / 10;
  m = n % 10;
  s = s + m;
  n = n / 10;
  m = n % 10;
  s = s + m;
  n = n / 10;
  printf ("%d", s);
}
```

while (n != 0)

WAP to find sum of 'n' digits number of digits:-

[OR]

```
int main ()
{ int n, sum = 0, m;
  printf ("Enter a number:");
  scanf ("%d", &n);
  while (n != 0)
  { m = n % 10;
    sum = sum + m;
    n = n / 10;
  }
  printf ("Sum of individual digits %d", sum);
}
```

Q:- WAP to reverse a number:-

```
-> main()
{
  int n, m, s=0;
  printf("enter n");
  scanf("%d", &n);
  m = n%10;
  s = s*10+m;
  n = n/10;
  m = n%10;
  s = s*10+m;
  n = n/10;
  m = n%10;
  s = s*10+m;
  n = n/10;
  printf("%d", s);
}
```

[OR]

```
-> main()
{
  int n, reverse=0, rem;
  printf("enter number
  to be reversed:");
  scanf("%d", &n);
  while(n!=0)
  {
    rem = n%10;
    reverse = reverse*10
              + rem;
    n/=10;
  }
  printf("The reversed
  number is: %d",
    reverse);
}
```