3. operators and Expressions

* Arithmetic operators.

+	addition or unary plus
-	subtraction or unary minus
*	multiplication
/	division
%	modulus division (remainder).

1) convert days into months and days.

```
#include<stdio.h>
main()
{
int months, days;
printf("Enter days\n");
scanf("%d", &days);
months = days/30;
days = days%30;
printf("months = %d days = %d", months, days);
}
```

* Relational operators.

To compare two quantities.

<	is less than
<=	is less than or Equal to
>	is greater than
>=	is greater than or Equal to
==	is Equal to
!=	is not Equal to

* logical operators

&&	logical AND
||	logical OR
!	NOT.

* Assignment operators.

Simple assignment operator

a = a+1	=>	a+=1
a = a-1	=>	a-=1
a = a*(n+1)	=>	a*=n+1
a = a/(n+1)	=>	a/=n+1
a = a%b	=>	a%=b

<- shorthand operator

2) find square of number using shorthand operator

```
#define N 100
#define A 2
main()
{
  int a;
  a = A;
  while(a < N)
    {
      printf("%d\n", a);
      a *= a;
    }
}
```

* Increment & decrement operators.

C allows 2 very useful operators not generally found in other languages are increment and decrement. Both are unary operators

++ → increment by 1
-- → decrement by 1

* conditional operator

* ternary operator pair "?:" is available in C

exp1 ? exp2 : exp3
↓ conditional ↓ true ↓ false

Eg: x = (a>b) ? a : b ;

(or)

```
if (a > b)
  x = a;
else
  x = b;
```

* Bitwise operators

&	bitwise AND
|	bitwise OR
^	bitwise XOR
<<	Shift left
>>	Shift Right

* Special operators.

comma operator

Sizeof operator

* arithmetic operators expressions

algebraic expression	c expression
a x b - c	a * b - c
(m+n)(x+y)	(m+n) * (x+y)
$\left(\frac{ab}{c}\right)$	a * b / c
$3x^2+2x+1$	3 * x * x + 2 * x + 1

* Evaluation of expressions

variable = expression;

Eg: x = a * b - c;

y = b / c * a;

3) Illustrations of evaluation of expressions:

```
main()
{
 float a,b,c,x,y,z;
 a = 9;
 b = 12;
 c = 3;
 x = a - b / 3 + c * 2 - 1;
 y = a - b / (3+c) * (2-1);
 z = a - (b / (3+c) * 2) - 1;
 printf("x = %f \n", x);
 printf("y = %f \n", y);
 printf("z = %f \n", z);
}
```

* precedence of arithmetic operators.

High priority * / %

Low priority + -

4) program for expression: a = 5 <= 8 && 6 != 5

```
#include <stdio.h>
#include <conio.h>
void main()
{
 int a;
 a = 5 <= 8 && 6 != 5;
 printf("%d", a);
 getch();
}
```

5) Sum of n terms of 1/n.

```
main()
{ float sum, n, term;
  int count = 1;
  sum = 0;
  printf("\n Enter value of n \n");
  scanf("%f", &n);
  term = 1.0/n;
  while(count <= n)
  {
    sum = sum + term;
    count++;
  }
  printf("\n Sum = %f \n", sum);
}
```

```c
/* rotate the variables*/
#include <stdio.h>
int main()
{
    int x, y, z, temp;
    printf("Enter values of x, y, z: ");
    scanf("%d %d %d", &x, &y, &z);
    temp = x;
    x = y;
    y = z;
    z = temp;
    printf("After rotation:\n");
    printf("x = %d\n", x);
    printf("y = %d\n", y);
    printf("z = %d\n", z);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./rotate
Enter values of x, y, z: 9 10 11
After rotation:
x = 10
y = 11
z = 9
```

```c
#include <stdio.h>
int main()
{
    float num;
    int intPart, rightmostDigit;
    printf("Enter a floating point number: ");
    scanf("%f", &num);
    intPart = (int)num;
    rightmostDigit = intPart % 10;
    if (rightmostDigit < 0)
    {
        rightmostDigit = -rightmostDigit;
    }
    printf("Rightmost digit of integral part: %d\n", rightmostDigit);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./float
Enter a floating point number: 98.76
Rightmost digit of integral part: 8
```

```c
/*two digits*/
#include <stdio.h>
int main()
{
    float num;
    int integralPart, twoDigits;
    printf("Enter a floating point number: ");
    scanf("%f", &num);
    integralPart = (int)num;
    twoDigits = integralPart % 100;
    if (twoDigits < 0)
    {
        twoDigits = -twoDigits;
    }
    printf("Two rightmost digits of integral part: %02d\n", twoDigits);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./two
Enter a floating point number: 12345.90
Two rightmost digits of integral part: 45
```

```c
/* area and perimeter of rectangle*/
#include <stdio.h>
int main()
{
    float length, width, area, perimeter;
    printf("Enter the length of the rectangle: ");
    scanf("%f", &length);
    printf("Enter the width of the rectangle: ");
    scanf("%f", &width);
    area = length * width;
    perimeter = 2 * (length + width);
    printf("Area of the rectangle = %.2f\n", area);
    printf("Perimeter of the rectangle = %.2f\n", perimeter);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./rec
Enter the length of the rectangle: 32
Enter the width of the rectangle: 45
Area of the rectangle = 1440.00
Perimeter of the rectangle = 154.00
```

```c
#include <stdio.h>
int main()
{
    float pp, aD, salvageValue;
    int years;
    printf("Enter purchase price: ");
    scanf("%f", &pp);
    printf("Enter years of service: ");
    scanf("%d", &years);
    printf("Enter annual depreciation: ");
    scanf("%f", &aD);
    salvageValue = pp - (aD * years);
    printf("Salvage Value = %.2f\n", salvageValue);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./value
Enter purchase price: 45
Enter years of service: 3
Enter annual depreciation: 13.3
Salvage Value = 5.10
```

```c
/*simple interest*/
#include <stdio.h>
int main()
{
    float p, r, t, si;
    printf("Enter Principal, Rate, Time: ");
    scanf("%f %f %f", &p, &r, &t);
    si = (p * r * t) / 100;
    printf("Simple Interest = %.2f\n", si);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./simple
Enter Principal, Rate, Time: 10000 5 36.5
Simple Interest = 18250.00
```

```c
/*Check whether one number is greater than another*/
#include <stdio.h>
int main()
{
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);
    if (a > b)
        printf("%d is greater than %d\n", a, b);
    else
        printf("%d is greater than %d\n", b, a);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./big
Enter two numbers: 8 9
9 is greater than 8
```

```c
/*Check whether two numbers are equal*/
#include <stdio.h>
int main()
{
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);
    if (a == b)
        printf("Both numbers are equal.\n");
    else
        printf("Numbers are not equal.\n");
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./equal
Enter two numbers: 7 7
Both numbers are equal.
supriya@ubuntu:~/Desktop/c/chp3$ ./equal
Enter two numbers: 1 2
Numbers are not equal.
```

```c
/*Check whether a number is positive, negative, or zero*/
#include <stdio.h>
int main()
{
    int n;
    printf("Enter a number: ");
    scanf("%d", &n);
    if (n > 0)
        printf("Positive number\n");
    else if (n < 0)
        printf("Negative number\n");
    else
        printf("Zero\n");
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./check
Enter a number: -9
Negative number
supriya@ubuntu:~/Desktop/c/chp3$ ./check
Enter a number: 8
Positive number
```

```c
#include <stdio.h>
int main() {
    int x = 5;
    printf("Initial x = %d\n", x);
    printf("Pre-increment (++x) = %d\n", ++x);
    printf("After pre-increment, x = %d\n", x);
    printf("Post-increment (x++) = %d\n", x++);
    printf("After post-increment, x = %d\n", x);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./inc
Initial x = 5
Pre-increment (++x) = 6
After pre-increment, x = 6
Post-increment (x++) = 6
After post-increment, x = 7
```

```c
/*Demonstrate pre-decrement and post-decrement*/
#include <stdio.h>
int main() {
    int x = 5;
    printf("Initial x = %d\n", x);
    printf("Pre-decrement (--x) = %d\n", --x);
    printf("After pre-decrement, x = %d\n", x);
    printf("Post-decrement (x--) = %d\n", x--);
    printf("After post-decrement, x = %d\n", x);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./dec
Initial x = 5
Pre-decrement (--x) = 4
After pre-decrement, x = 4
Post-decrement (x--) = 4
After post-decrement, x = 3
```

```c
/*Demonstrate shorthand operator*/
#include <stdio.h>
int main()
{
    int x, y;
    printf("Enter x and y: ");
    scanf("%d %d", &x, &y);
    x += y;
    printf("Result of x += y is %d\n", x);
    x-=y;
    printf("Result of x-=y is %d\n",x);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./short
Enter x and y: 3 4
Result of x += y is 7
Result of x-=y is 3
```

```c
/*Demonstrate bitwise*/
#include <stdio.h>
int main()
{
    int a, b;
    printf("Enter two integers: ");
    scanf("%d %d", &a, &b);
    printf("%d & %d = %d\n", a, b, a & b);
    printf("%d | %d =%d\n",a,b,a|b);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./bit
Enter two integers: 1 3
1 & 3 = 1
1 | 3 =3
```

```c
/*shift operators*/
#include <stdio.h>
int main() {
    int a, n;
    printf("Enter an integer and number of positions to: ");
    scanf("%d %d", &a, &n);
    printf("%d << %d = %d\n", a, n, a << n);
    printf("%d >> %d =%d\n",a,n,a>>n);
    return 0;
}
```

```c
/*Evaluate the expression ((a + b) * c - d) / e*/
#include <stdio.h>
int main() {
    float a, b, c, d, e, result;
    printf("Enter values of a, b, c, d, e: ");
    scanf("%f %f %f %f %f", &a, &b, &c, &d, &e);
    if (e != 0)
    {
        result = ((a + b) * c - d) / e;
        printf("Result of ((a+b)*c - d)/e = %.2f\n", result);
    } else
    {
        printf("Division by zero is not allowed.\n");
    }
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./exp
Enter values of a, b, c, d, e: 1 2 3 4 5
Result of ((a+b)*c - d)/e = 1.00
```

```c
/*even or odd*/
#include <stdio.h>
int main()
{
    int num;
    printf("Enter an integer: ");
    scanf("%d", &num);
    if (num % 2 == 0)
        printf("%d is even\n", num);
    else
        printf("%d is odd\n", num);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./test
Enter an integer: 46
46 is even
```

```c
#include <stdio.h>
int main() {
    int marks1, marks2;
    float average;
    printf("Enter marks of two subjects: ");
    scanf("%d %d", &marks1, &marks2);
    average = (float)(marks1 + marks2) / 2;
    printf("Average marks = %.2f\n", average);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./mark
Enter marks of two subjects: 45 67
Average marks = 56.00
```

```c
#include <stdio.h>
int main()
{
    float a, b, c;
    float sum, average;
    float largest, smallest;
    printf("Enter three numbers: ");
    scanf("%f %f %f", &a, &b, &c);
    sum = a + b + c;
    average = sum / 3;
    largest = a;
    if (b > largest)
            largest = b;
    if (c > largest)
            largest = c;
    smallest = a;
    if (b < smallest)
            smallest = b;
    if (c < smallest)
            smallest = c;
    printf("Sum = %.2f\n", sum);
    printf("Average = %.2f\n", average);
    printf("Largest = %.2f\n", largest);
    printf("Smallest = %.2f\n", smallest);
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./abc
Enter three numbers: 4 5 6
Sum = 15.00
Average = 5.00
Largest = 6.00
Smallest = 4.00
```

```c
#include <stdio.h>
int main()
{
    int m, n;
    printf("Enter two integers (m and n): ");
    scanf("%d %d", &m, &n);
    if (n == 0) {
        printf("Cannot divide by zero.\n");
    }
    else {

        if (m % n == 0)
            printf("%d is a multiple of %d\n", m, n);
        else
            printf("%d is not a multiple of %d\n", m, n);
    }
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./multiple
Enter two integers (m and n): 35 46
35 is not a multiple of 46
```

```c
#include <stdio.h>
#include <math.h>
int main() {
    int angle;
    double radians, sineValue, cosineValue;
    printf("Angle(deg)\tSin\t\tCos\n");
    printf("-------------------------------\n");
    for (angle = 0; angle <= 180; angle += 15)
    {
        radians = angle * M_PI / 180.0;
        sineValue = sin(radians);
        cosineValue = cos(radians);
        printf("%3d\t\t%.4f\t%.4f\n", angle, sineValue, cosineValue);
    }
    return 0;
}
```

```
supriya@ubuntu:~/Desktop/c/chp3$ ./sine
Angle(deg)      Sin             Cos
-------------------------------
  0             0.0000  1.0000
 15             0.2588  0.9659
 30             0.5000  0.8660
 45             0.7071  0.7071
 60             0.8660  0.5000
 75             0.9659  0.2588
 90             1.0000  0.0000
105             0.9659  -0.2588
120             0.8660  -0.5000
135             0.7071  -0.7071
150             0.5000  -0.8660
165             0.2588  -0.9659
180             0.0000  -1.0000
```