Bitwise operators in C:

1] & = Bitwise AND: The output is 1 if, the corresponding bits of two operands is 1

2] | = Bitwise OR: The output is 1 if, at least one of the corresponding bits of two operands is 1

3] ^ = Bitwise XOR: The output is 1 if, corresponding bits are opposite.

4] ~ = Bitwise compliment: Changes 0 to 1, 1 to 0. (For int N bitwise complement will be –(N+1))

5] << = Shift left: Shifts left by certain number of specified bits.

6] >> = Shift right: Shifts right by certain number of specified bits.

Ternary operators in C:

It helps to think of the ternary operator as a shorthand way or writing an if-else statement.

Condition ? value_if true : value_if false

Calculator Code(using +,-,*,/,%):

```
#include<stdio.h>

void main()

{

   int a,b;

   printf("Enter a: ");

   scanf("%d",&a);

   printf("Enter b: ");

   scanf("%d",&b);

   printf("Sum of %d and %d is %d\n",a,b,a+b);

   printf("Difference of %d and %d is %d\n",a,b,a-b);

   printf("Multiplication of %d and %d is %d\n",a,b,a*b);

   printf("a divided by b: %d\n",a/b);

   printf("Remainder when a divides b: %d\n",a%b);

}
```