1) Printing the binary representation of any Number.

```
void pr_binary(int num){
   for(int i=10;i>=0;i--) cout<<((num>>i)&1);
   cout<<endl;
}
```

//Update :You can also represent any number in its binary form as;

```
cout<<bitset<const_length>(number);
```

2) checking if the ith bit is set or not.

```
if((a&(1<<i))!=0) cout<<"set"<<endl;
  // check if set or not;
  else cout<<"Not set"<<endl;
```

3) Counting the number of set bits

```
  int ans=0;
  for (int i=31; i>=0;--i)
  {
    if((a&(1<<i))!=0) ans++;
  }
  dis(ans);
```

//Even though the inbuilt function is also there.

```
  cout<<__builtin_popcountll((1ll<<35)-1);
```

4) Some other important operations.

```
  pr_binary(a | (1<<i));
  // set that ith bit;


  pr_binary(a&(~(1<<i)));
  // unset the ith bit;
  pr_binary(a ^ (1<<i));
  // toggle the ith bit from set to unset and vice-versa;
```

5) For getting the count of odd,even to a number n;

```
   int n,od=0,ev=0;
   cin>>n;
   for(int i=1;i<=n;i++){
    if(i&1) od++;
    else ev++;
  }
  cout<<"Count of Odd"<<od<<endl;
  cout<<"Count of Even"<<ev<<endl;
```

6) Dividing or multiplying any number by two

//Although the arithmetic operations are fast ,but by bits manipulation we can make them //more faster.

```
  int n=5;
  n=n>>1;
  // divide by two
  dis(n);
  n=n<<1;
  // multiply by two
```

7) Some cool operations and playing with Characters

```
   for(char c='A';c<='Z';c++){
     cout<<c<<" ";
     pr_binary(int(c));
   }
   for(char c='a';c<='z';c++){
     cout<<c<<" ";
     pr_binary(int(c));
   }
   //difference between upper case letter and lower case letter binary is that
   //in upper case letter 5th bit!=1;
   //in lower case letter 5th bit =1;
   cout<<char('A'|(1<<5))<<endl;
   //in lower case;


   cout<<char('a'&(~(1<<5)))<<endl;
   //in upper case;


   //actually char of 1<<5 is _(space);
   //take any upper case letter and its ||  with space will get the corresponding lower case letter;


   cout<<char('C'|' ')<<endl;
  // will make it small c
//take any lower case letter and its || with _(underscore) will get the corresponding upper //case
letter;`


   cout<<char('c'&'_')<<endl;
  // will make it capital C
```

8) Swap with XOR.

```
 int a=4;
  int b=5;
  a=a^b;
  b=b^a;
  a=a^b;
  // cout<<a<<" "<<b;
```

9) Checking if a number is the power of two.

```
  int n=16;
  n&(n-1)?dis("NO"):dis("YES");


  //Update : this will not work for n==0;
  for n=0;
```

```
//we can have the function
bool check_power_of_two(int num){
  return n && !(n&(n-1));
}
```

10) For clearing the set bits upto ith bit

```
 int i=4;
//clearing upto 5 the place;
int a=59;
int b=(a&(~((1<<(i+1))-1)));
//clearing the lsb upto ith bit;
pr_binary(b);


i=3;
int c=(a&((1<<(i+1))-1));
//clearing the msb upto ith bit;
pr_binary(c);
```

If you find curious about bit manipulations, there other techniques

- Find max/min without branching
- Negative a number without branching
- Find absolute value without branching
- Set a specific bit: set the ppth bit to x∈x∈ {0,10,1} without using if-branch
- Find square root
- Find cube root
- Find logaritm
- Reverse all bit
- Reverse bits from bit LL to bit RR
- Fast modulus for special cases
- Next higher power of 2
- Prev smaller power of 2
- Interleave bits

- Next/Prev lexicographical bit permutation
- Enumerating submask
- Enumerating masks and submask in lexicographical order