```
<include <bits/stdc++.h#

;using namespace std
*/
program to count number of bits in number , bits counted is the 1 bits which in
other words light bits or positive bit or on bit
like 26 in binary representation  = 11010 which is = 2^1 + 2^3 + 2^4 = 2 + 8 +
,16 = 26
so we need to count light or positive bits in any number like for example above
is 3 bits
bits work from right to left so first bit for number 26 is
0 ^ 2 = 0
1 ^ 2 = 1
2 ^ 2 = 0
3 ^ 2 = 1
4 ^ 2 = 1
on bit in this number are 3
f is about function
i is about integer data type

/*


int f_count_bits_in_number(int i_msk_number)
}
;int i_return_bits_count  = 0
while(i_msk_number)
}
        /*
            next lines is about
            number & 1 is a bit wise and & operator
            which help you to know the last bit of any number is turn on or off
which is last bit is 0 off or 1 on
            so for example 26 is 11010 so last bit is 0 so you check for last bit
with 1
            because if last bit is 0 will ignore the count otherwise which 1 will
count it as on bit
            you should know
            if(0&0) is fals
            for more details check table below it is about logical operator

            A   B       !A  A&B  A|B
            0     0     1   0  0
          0   1   1   0  1
          1       0     0   0  1
            1     1     0   1  1
            -----------------------------

        */
        if(i_msk_number & 1) ++i_return_bits_count;

        /*
            next line is
             right shift operator which help you to remove last bit from msk by
devide by 2 so
             11010 >> 1 is equal to 1101 and so the law is 2^n like i_msk_number >>
1 means i_msk_number / 2^1
             if you say i_msk_number >> 2 whill be i_msk_number / 2 ^ 2 and so on
             you can read about right shift and left shift operator
             its save your memory time so mush may by 40 speead time than /
operator
             and you can use bitwise operator with your condition also like if( 26
& 1 == 0) to check is number is even or odd
        */
        i_msk_number >>= 1;
    }
    return i_return_bits_count;
}

int main()
{
    int i_msk_number;
    /*
        you can comment the loop and read by scanf or cin
        any number you need and pass it as below or use printf in the loop or use
code below to read

            scanf("%d", &i_msk_number);
            printf("%d\n", f_count_bits_in_number(i_msk_number));
    */
    for(int i=0; i < rand() % 100000;++i) // i looped on random value also
    {
        i_msk_number = rand() % 1000;
        printf("%d\n", f_count_bits_in_number(i_msk_number));
    }
    return 0;
}
```