406410114 UVA10019 Funny Encryption Method

```
//使用Java的考生請注意，最外層的類別(class)需命名為 main 。
//如果遇到超乎想像的狀況，請更改編譯器試看看!! 各編譯器特性不同!!
//預設測資、隨機測資、固定測資是用來幫助除錯用的。批改時，只看暗中測資是否通過!!

// input n
//output (dec to binary 1's num)  and  ( hex to binary 1's num  )
#include <bits/stdc++.h>
using namespace std;

int dec_to_bin_one_num( int dec,vector <int> binary ){
        int i,j,k,one_num = 0;
        for ( i = binary.size(); i>=0 ;i--  ){
                //cout << "dec/binary[i]: " << dec/binary[i] <<endl;
                if ( dec/binary[i] == 1 ){
                        one_num++;
                        dec %= binary[i];
                }//if
        }//for

        return one_num;
}//dec_to_bin_one_num



int main(){
        int i,j,k,n,temp;
        vector <int> binary,hex;
        temp = 1;
        for ( i = 1 ;i <20;i++ ){
                binary.push_back( temp );
                temp *=2;
        }//for


        cin >>n;
        while (n--){
                int num1,hex_temp = 0;
                cin >> num1;
                cout << dec_to_bin_one_num( num1,binary )<< " ";

                //cout <<"hex_temp: "<<hex_temp<<endl;
                hex_temp += (num1/10000) *(16*16*16*16);
                num1 %= 10000;
                //cout <<"hex_temp: "<<hex_temp<<endl;
                hex_temp += (num1/1000) *(16*16*16);
                num1 %= 1000;
                //cout <<"hex_temp: "<<hex_temp<<endl;
                hex_temp += (num1/100) *(16*16);
                num1 %= 100;
                //cout <<"hex_temp: "<<hex_temp<<endl;
                hex_temp += (num1/10) *(16);
                num1 %= 10;
                //cout <<"hex_temp: "<<hex_temp<<endl;
                hex_temp+= (num1);



                //cout << "hex_temp should is : " <<2*16*16+6*16+5 <<endl;
                //cout <<"hex_temp: "<<hex_temp<<endl;
                cout << dec_to_bin_one_num( hex_temp,binary ) << endl;

        }//while
```

```
        return 0;
}//main


//== 以上是自[瘋狂雲端]下載的資料 請自行剪貼到各檔中進行寫作
/*
//使用Java的考生請注意，最外層的類別(class)需命名為 main 。
//如果遇到超乎想像的狀況，請更改編譯器試看看!! 各編譯器特性不同!!
//預設測資、隨機測資、固定測資是用來幫助除錯用的。批改時，只看暗中測資是否通過!!

//*/
```