

# Sum vs XOR

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

using namespace std;

string ltrim(const string &);
string rtrim(const string &);

/*
 * Complete the 'sumXor' function below.
 *
 * The function is expected to return a LONG_INTEGER.
 * The function accepts LONG_INTEGER n as parameter.
 */

long sumXor(long n) {
    if (n==0) {
        return 1;
    }
    long count=0;
    while (n>0) {
        if (n%2==0) {
            count=count+1;
        }
        n=n/2;
    }
    long ans=1;
    for (long i=0; i<count; i++) {
        ans=ans*2;
    }
    return ans;
}

int main()
{
    ofstream fout(getenv("OUTPUT_PATH"));

    string n_temp;
    getline(cin, n_temp);

    long n = stol(ltrim(rtrim(n_temp)));

    long result = sumXor(n);
```

```

    fout << result << "\n";

    fout.close();

    return 0;
}

string ltrim(const string &str) {
    string s(str);

    s.erase(
        s.begin(),
        find_if(s.begin(), s.end(), not1(ptr_fun<int,
int>(isspace)))
    );

    return s;
}

string rtrim(const string &str) {
    string s(str);

    s.erase(
        find_if(s.rbegin(), s.rend(), not1(ptr_fun<int,
int>(isspace))).base(),
        s.end()
    );

    return s;
}

```