Flipping bits

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

using namespace std;

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

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

long flippingBits(long n) {
    return (uint) ~n;
}

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

    string q_temp;
    getline(cin, q_temp);

    int q = stoi(ltrim(rtrim(q_temp)));

    for (int q_itr = 0; q_itr < q; q_itr++) {
        string n_temp;
        getline(cin, n_temp);

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

        long result = flippingBits(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;
}
```