

CUIT CTF WriteUp-小白和小黑的故事

原创

[RickGray](#) 于 2014-05-22 16:19:13 发布 1250 收藏

分类专栏: [CTF纪实](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/u013565525/article/details/26599585>

版权



[CTF纪实](#) 专栏收录该内容

10 篇文章 0 订阅

订阅专栏

此题考查基本的ACM知识-矩阵快速幂, 下面直接上题目描述

有一颗神秘的白球, 有一种神秘的性质:

1. 白色的球, 每过一年会分裂成3个白球和1个黑球;

2. 黑色的球, 每过一年会分裂成1个白球和3个黑球;

问题: 求n年之后 (10^{15} 左右), 一共会有多少个白球? (结果对1000000007取模)

分析:

由题目信息可知, 此题类似于求费波拉契数列 (第n项, n很大) 的问题 (通过矩阵快速幂的方法), 分析题目后可以得到推出如下结论:

假设第n年有白球x个, 黑球y个, 那么第n+1年就有白球 $(3x+y)$ 个, 黑球 $(3y+x)$ 个, 转化成矩阵表示, 就有

$$F(n) = \begin{pmatrix} x \\ y \end{pmatrix}, F(n+1) = \begin{pmatrix} 3x+y \\ x+3y \end{pmatrix} \pmod{1000000007}, \leftarrow$$

$$F(n+1) = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} * F(n) \pmod{1000000007}, \text{ 令 } A = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}, \leftarrow$$

$$\text{由因为 } F(1) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \leftarrow$$

所以通过递推可得, 第n年的球数为

$$F(n) = A^{n-1} * F(1) \pmod{1000000007}:$$

从上述推导可以看出求第n年的白球数直接可以转化为求A的n-1次幂的问题

一个矩阵的超高次幂的计算可以直接用矩阵快速幂的算法来计算, 具体算法这里就不描述了, 下面是程序的测试截图

```

C:\Users\Meos\Desktop\Syclover\BlackWhiteSplit\work.exe
Round 1 After 363902530924078 year(s)
Input the number of white balls:381841495
Round 2 After 364192533824107 year(s)
Input the number of white balls:1244553
Round 3 After 364452536424133 year(s)
Input the number of white balls:195431212
Round 4 After 364852540424173 year(s)
Input the number of white balls:903754084
Congratulations! Key:
1.#1.#20#32#54#86#138#21:#34<#55>#89#>4B#G3D#<1F#98

C:\Users\Meos\Desktop\Syclover\BlackWhiteSplit\Demo.exe
wd: 2
mod: 1000000007
Input exp: 363902530924078
White Ball: 381841495

wd: 2
mod: 1000000007
Input exp: 364192533824107
White Ball: 1244553

wd: 2
mod: 1000000007
Input exp: 364452536424133
White Ball: 195431212

wd: 2
mod: 1000000007
Input exp: 364852540424173
White Ball: 903754084

wd: 2
mod: 1000000007
Input exp:

```

cpp代码如下

```

#include <cstdio>
#include <iostream>
#include <cstring>
using namespace std;

const int NN = 2;

typedef struct node
{
    long long matrix[NN][NN];
} Matrix;
Matrix a,sa,unit;
long long n, m, k, ex;

Matrix Mul(Matrix a,Matrix b) //矩阵乘法(%m)
{
    Matrix c;
    for (int i=0; i<n; i++)
        for (int j=0; j<n; j++) {
            c.matrix[i][j]=0;
            for (k=0; k<n; k++)
                c.matrix[i][j]+=a.matrix[i][k]*b.matrix[k][j];
            c.matrix[i][j]%=m;
        }
    return c;
}

Matrix Cal(long long exp) //矩阵快速幂
{
    Matrix p,q;
    p = a;
    q = unit;
    while (exp!=1) {
        if (exp&1) {
            exp--;
            q=Mul(p,q);
        }
        else {
            exp>>=1;

```

```

        p=Mul(p,p);
    }
}
return Mul(p,q);
}

int main() {
    int ww = 5;
    while(ww) {
        n = 2;
        printf("矩阵阶: %lld\n", n);
        m = 1000000007;
        printf("模数: %lld\n", m);
        printf("请输入年数: ");
        scanf("%lld",&ex);
        //printf("exp: %lld\n", ex);

        a.matrix[0][0] = 3;
        a.matrix[0][1] = 1;
        a.matrix[1][0] = 1;
        a.matrix[1][1] = 3;

        for (int i=0; i<n; i++)
            for (int j=0; j<n; j++)
                {
                    unit.matrix[i][j]=(i==j);
                }
        sa=Cal(ex);

        printf("White Ball: %lld\n\n", sa.matrix[0][0]);
        ww--;
    }

    getchar();
    return 0;
}

```