**#include <iostream>**

**矩阵快速幂(求余)**

**#include <stdio.h>**

**#include <string.h>**

**using namespace std;**

**#define LL long long**

**#define MOD 1000000007**

**struct Matrix**

**{**

**LL m[3][3];**

**};**

**LL n; //计算 f[n]=f[n-1+f[n-3] 的f[n]**

**Matrix Mult(Matrix &s,Matrix &p)**

**{**

**Matrix re;**

**for (int i=0;i<3;i++)**

**{**

**for (int j=0;j<3;j++)**

**{**

**re.m[i][j]=0;**

**for (int k=0;k<3;k++)**

**re.m[i][j]=(re.m[i][j]+ s.m[i][k]\*p.m[k][j])%MOD;**

**}**

**}**

**return re;**

**}**

**LL func(LL n)**

**{**

**if (n<4) //n太小的情况**

**{**

**LL num[4]={0,2,3,4};**

**return num[n];**

**}**

**Matrix s,b;**

**b.m[0][0]=1,b.m[0][1]=0,b.m[0][2]=1;**

**b.m[1][0]=1,b.m[1][1]=0,b.m[1][2]=0;**

**b.m[2][0]=0,b.m[2][1]=1,b.m[2][2]=0;**

**memset(s.m,0,sizeof(s.m));**

**for (int i=0;i<3;i++) s.m[i][i]=1;**

**LL x = n-3;**

**while(x){**

**if (x&1) s = Mult(s,b);**

**b = Mult(b,b);**

**x/=2;**

**}**

**return (s.m[0][0]\*4+s.m[0][1]\*3+s.m[0][2]\*2)%MOD;**

**}**