

```

#include<cstring>
#include<cstdio>
#include<iostream>
#include<algorithm>
#include<vector>
#include<cmath>
#include<map>
#define l(x) (x<<1)
#define r(x) ((x<<1)+1)
#define IL inline
#define reg register
#define LL long long
#define MOD 10000007
#define INF 0x3f3f3f3f
using namespace std;

LL n,len,l;
LL z[75],f[75][75];

IL int Abs(int x){return (x<0)?-x:x;}
IL void Swap(int &a,int &b){a^=b^=a^=b;}
IL int Min(int a,int b){return (a<b)?a:b;}
IL int Max(int a,int b){return (a>b)?a:b;}

IL int read(){
    int p=0,f=1;    char    c=getchar();
    while (c<48||c>57) {if (c=='-')    f=-1;    c=getchar();}
    while (c>=48&&c<=57)    p=(p<<1)+(p<<3)+c-48,c=getchar();
    return p*f;
}

IL LL Mi(LL x,LL y){
    LL p=x,t=1,Res=1;
    for (;t<=y;(t&y)?Res=(Res*p)%MOD:0,p=(p*p)%MOD,t<<=1);
    return Res;
}

IL LL Dfs(int pos,int pre,int lead,int limit,int s){
    LL sum=0,up=0,i=0;
    if (pos>len)    return (s==l);
    if (!lead && !limit && f[pos][s]!=-1)    return f[pos][s];
    up=(limit)?z[len-pos+1]:1;
    for (i=0;i<=up;i++){
        if (!i && lead) sum+=Dfs(pos+1,0,1,limit&&(i==up),s);
        else if (i && lead) sum+=Dfs(pos+1,i,0,limit&&(i==up),s+1);
        else sum+=Dfs(pos+1,i,0,limit&&(i==up),s+(i==1));
    }
    return (!lead && !limit)?f[pos][s]=sum:sum;
}

IL LL Cal(LL x){

```

```
LL Ans=1,sum=0,i=0;
len=0;
while (x){
    z[++len]=x%2;    x/=2;
}
for (i=1;i<=len;i++){
    memset(f,-1,sizeof(f));
    l=i;
    sum=Dfs(1,0,1,1,0);
    Ans=(Ans*Ml(l,sum))%MOD;
}
return Ans;
}

int main(){
#ifdef __Marvolo
    freopen("zht.in","r",stdin);
    freopen("zht.out","w",stdout);
#endif
    scanf("%lld",&n);
    cout<<Cal(n)<<endl;
    return 0;
}
```

From:

<https://wiki.cvbbacm.com/> - **CVBB ACM Team**

Permanent link:

<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:t1e233:code2>



Last update: **2020/05/10 15:01**