

```
1 #include "../bits/stdc++.h"
2 // verified: http://judge.u-aizu.ac.jp/onlinejudge/review.jsp?rid=3613584#1
3
4 template <typename T>
5 using matrix = std::vector<std::vector<T>>;
6
7 constexpr long long mod = 1e9+7;
8
9 template <typename T>
10 matrix<T> modmul(const matrix<T> &l, const matrix<T> &r)
11 {
12     matrix<T> ret(l.size(), std::vector<T>(r[0].size()));
13     for (int i = 0; i < (int)l.size(); i++)
14     {
15         for (int k = 0; k < (int)r.size(); k++)
16         {
17             for (int j = 0; j < (int)r[0].size(); j++)
18             {
19                 ret[i][j] += l[i][k] * r[k][j] % mod;
20                 ret[i][j] %= mod;
21             }
22         }
23     }
24     return ret;
25 }
26
27 template <typename T>
28 std::vector<T> modmul(const matrix<T> &a, const std::vector<T> &x)
29 {
30     const int n = a.size();
31     const int m = x.size();
32     std::vector<T> ret(n);
33     for (int i = 0; i < n; i++)
34     {
35         for (int j = 0; j < m; j++)
36         {
37             ret[i] += a[i][j] * x[j] % mod;
38             ret[i] %= mod;
39         }
40     }
41     return ret;
42 }
43
44 template <typename T>
45 matrix<T> eye(const int n)
46 {
47     matrix<T> ret(n, std::vector<T>(n));
48     for (int i = 0; i < n; i++)
49         ret[i][i] = 1;
50     return ret;
51 }
52
53 template <typename T>
54 matrix<T> modpow(matrix<T> x, long long n)
55 {
56     auto ret = eye<T>(x.size());
57     while (n)
58     {
59         if (n & 1)
60             ret = modmul(ret, x);
61         x = modmul(x, x);
62         n /= 2;
63     }
64     return ret;
65 }
66
```