/\*

PROG: cowfood

LANG: C++

ID: hayk.sa1

\*/

#include <stdio.h>

#define mod 100000000

short fib;

short n, m;

int d[12][400];

short field[12];

short number[400];

bool row[12][400];

bool can[400][400];

void read()

{

short l, i, j;

scanf("%hd%hd", &n, &m);

for (i = 0; i < n; i++)

for (j = 0; j < m; j++)

{

scanf("%hd", &l);

field[i] = (field[i]<<1)|l;

}

}

void build()

{

short l, i, j;

short f[13];

f[0] = 1;

f[1] = 2;

for (i = 2; i <= m; i++)

f[i] = f[i-1]+f[i-2];

fib = f[m];

for (l = i = 0; l < fib; i++)

{

for (j = 1; j < m && !(((i>>j)&1) && ((i>>(j-1))&1)); j++);

if (j == m)

number[l++] = i;

}

for (i = 0; i < fib; i++)

for (j = 0; j < fib; j++)

{

for (l = 0; l < m && !(((number[i]>>l)&1) && ((number[j]>>l)&1)); l++);

can[i][j] = (l == m);

}

for (i = 0; i < n; i++)

for (j = 0; j < fib; j++)

{

for (l = 0; l < m && !(((field[i]>>l)&1) == 0 && ((number[j]>>l)&1)); l++);

row[i][j] = (l == m);

}

for (i = 0; i < fib; i++)

d[0][i] = (int)(row[0][i]);

}

int solve()

{

int ans;

short i, j, k;

for (i = 1; i < n; i++)

for (j = 0; j < fib; j++)

if (row[i][j])

{

for (k = 0; k < fib; k++)

if (row[i-1][k] && can[j][k])

d[i][j] = (d[i][j]+d[i-1][k])%mod;

}

for (ans = i = 0; i < fib; i++)

ans = (ans+d[n-1][i])%mod;

return ans;

}

int main()

{

freopen("cowfood.in", "r", stdin);

freopen("cowfood.out", "w", stdout);

read();

build();

printf("%d\n", solve());

return 0;

}