#include<cmath>

#include<cstdio>

#include<iostream>

#include<string>

#include<algorithm>

#include<queue>

using namespace std;

int winN, winC, PeoN, ChekN;

struct {

int cons[1005] = { 0 };

int init[1005] = { 0 };

int time[1005] = { 0 };

int front = 1;

}peo;

int time\_jiong = 480;

queue<int>win[21];

void record();

void cipher();

void lookup();

int main() {

cin >> winN >> winC >> PeoN >> ChekN;

record();

cipher();

lookup();

return 0;

}

void record() {

for (int i = 1; i <= PeoN; i++)

{

cin >> peo.cons[i];

peo.init[i] = peo.cons[i];

}

for (int i = 1; i <= winN\*winC; i++)

{

int temp = i%winN;

win[temp].push(i);

peo.front++;

}

}

void cipher() {

while (time\_jiong<9000)

{

int flag = 0;

for (int i = 0; i < winN; i++)

{

if (win[i].empty() == 0)

flag = 1;

}

if (!flag)

break;

time\_jiong++;

for (int i = 0; i < winN; i++)

{

if (!win[i].empty())

{

int first = win[i].front();

peo.cons[first]--;

if (peo.cons[first] == 0)

{

if(time\_jiong-peo.init[first]<1020)

peo.time[first] = time\_jiong;

win[i].pop();

if (peo.front != PeoN + 1&&time\_jiong<1020)

win[i].push(peo.front++);

}

}

}

}

return;

}

void lookup() {

for (int i = 0; i < ChekN; i++)

{

int temp;

cin >> temp;

int t\_time = peo.time[temp];

if (t\_time != 0)

{

int hour = peo.time[temp] / 60, minute = peo.time[temp] % 60;

if (hour < 10 && minute < 10)

cout << "0" << hour << ":0" << minute << endl;

else if (hour < 10)

cout << "0" << hour << ":" << minute << endl;

else if (minute < 10)

cout << hour << ":0" << minute << endl;

else

cout << hour << ":" << minute << endl;

}

else

cout << "Sorry" << endl;

}

}