#include<bits/stdc++.h>

using namespace std;

void solve(int n, int arr[][10000]){

stack<int> st;

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

st.push(i);

}

while(st.size()!=1){

int v1 = st.top();

st.pop();

int v2 = st.top();

st.pop();

if(arr[v1][v2]==1){

st.push(v2);

}

else{

st.push(v1);

}

}

int idx = st.top();

bool flag = false;

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

if(i!=idx){

if(arr[i][idx]==0 || arr[idx][i]==1){

cout<<"none";

return;

}

}

}

cout<<idx<<endl;

}

int main(){

int n;

cin>>n;

int arr[n][10000];

string str[n];

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

cin>>str[i];

}

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

string s = str[i];

for(int j=0;j<n;j++){

arr[i][j] = s[j]-'0';

}

}

solve(n,arr);

}

----------------------------------my first approach-------------------------------------------------------------------

#include<bits/stdc++.h>

using namespace std;

void solve(int n, int arr[][10000]){

stack<int> st;

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

st.push(i);

}

while(st.size()!=1){

int v1 = st.top();

st.pop();

int v2 = st.top();

st.pop();

if(arr[v1][v2]==1){

st.push(v2);

}

else{

st.push(v1);

}

}

int idx = st.top();

bool flag = false;

for(int row=st.top();row<n;row++){

if(arr[idx][row]==0){

flag = true;

}

else{

flag = false;

break;

}

}

for(int col=st.top();col<n && col!=idx ;col++){

if(arr[col][idx]==1){

flag = true;

}

else{

flag = false;

break;

}

}

if(flag){

cout<<idx<<endl;

}

else{

cout<<"none"<<endl;

}

}

int main(){

int n;

cin>>n;

int arr[n][10000];

string str[n];

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

cin>>str[i];

}

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

string s = str[i];

for(int j=0;j<n;j++){

arr[i][j] = s[j]-'0';

}

}

solve(n,arr);

}