//Given an integer n find the factor of n for multiple test cases t

//This is The Coding Area

#include<bits/stdc++.h>

using namespace std;

int main()

{

long long int n,square,i,j;

int t;

int a;

int b;

int c;

int d;

cin>>t;

while(t--)

{

cin>>n;

square=sqrt(n);

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

{

if(n%i==0)

{

cout<<i<<" ";

}

}

for(i=square;i>=1;i--)

{

if(n%i==0)

{

cout<<n/i<<" ";

}

}

cout<<endl;

}

}

// Print the given pattern for given n

//if n==3

//10203010011012

// 4050809

// 607

//if n==4

//1020304017018019020

// 50607014015016

// 809012013

// 10011

#include<bits/stdc++.h>

using namespace std;

void shivam()

{

int a=4,b=7;

int c=a+b;

c=a\*b;

return ;

}

int main()

{

shivam();

int n,i,j,k,space=0,first=1,second;

cin>>n;

int m=n;

second=n\*n+1;

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

{

for(j=1;j<=space;j++)

cout<<" ";

for(k=1;k<=m;k++)

{

cout<<first<<"0";

first++;

}

for(k=1;k<=m;k++)

{

if(k==m)

cout<<second;

else

cout<<second<<"0";

second++;

}

cout<<endl;

m--;

second-=2\*m+1;

space++;

}}

**Question :**

Given an array of n elements and a number m, we need to find all distinct pairs existing in the array whose pair sum is divisible by the given number m and then print the total number of such pairs. Distinct pairs means (1, 2) and (2, 1) are the same.

**Question :** there are switches that can be controlled by a remote , n such switches are connected in series and at the end of there is a bulb , the remote has a button that toggle the state of the switch (on to off and vice versa) , but the remote only works in switches which have a electric connection to them , A switch is connected when when all the previous switches are turned ON, and the bulb is on when all the switches are ON , At first all the switches are off and only the first switch is connected ,

Input- N (number of switches ) ,1 < N < 30  
K(number of times the button is pressed) 1 < k < 10^7 output – 1 (if the bulb is on) , -1 (if it’s not).

**Question :** there are N employee sitting in consecutive cubicles , we have to send a few of them to onsite , but each time we send one employee onsite , his cubicle becomes empty , now the other employees from both side of that empty cubicle stops working until they are given a gift .  
the gifts are given in both sides of the empty cubicle until we reach the end or found another cubicle ,

Input – number of cubicle , and index numbers of people to be sent  
output- min number of gifts needed



