A [narcissistic number](http://en.wikipedia.org/wiki/Narcissistic_number) is a number that is equal to the sum of its digits each raised to the power equal to the total number of digits. For example, 153 is a narcissistic number, since 13 + 53 + 33 = 153.

Given the numbers start and finish, return all*narcissistic numbers* in range [start, finish](inclusive).

**Example**

For start = 100 and finish = 200, the output should be  
NarcissisticNumber(start, finish) = [153].

153 is the only number in range [100, 200] which is*narcissistic*.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] integer start**

*Constraints:*  
1 ≤ start ≤ finish.

* **[input] integer finish**

*Constraints:*  
start ≤ finish ≤ 104.

* **[output] array.integer**

A sorted array of *narcissistic numbers* betweenstart and finish (inclusive).

<https://codefights.com/challenge/3eZwiyP6btWXDxohS/main>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static bool isNarcissistic(int n)

{

int sum = 0;

string nstr =n.ToString();

for (int i = 0; i < nstr.Length; i++)

{

sum += (int)Math.Pow(int.Parse(nstr[i].ToString()), nstr.Length);

}

if (sum == n)

{

return true;

}

return false;

}

static int[] NarcissisticNumber(int start, int finish)

{

List<int> narc = new List<int>();

for (int i = start; i <= finish; i++)

{

if (isNarcissistic(i))

{

narc.Add(i);

}

}

return narc.ToArray();

}

static void Main(string[] args)

{

foreach (int elem in NarcissisticNumber(100, 200))

{

Console.Write(elem + " ");

}

Console.ReadLine();

}

}

}