You are given two integers L and R. What single digit occurs the most often in all the numbers between them (inclusive)? If there is more than one solution, return the smallest one.

* **[input] integer L**
  + 1 ≤ L ≤ 107.
* **[input] integer R**
  + 1 ≤ L ≤ R ≤ 107.
* **[output] integer**
  + The digit which occurs most often.

<https://codefights.com/challenge/2cZ5oB8MczpXFm9tR>

#include <iostream>

#include <conio.h>

#include <stdio.h>

int digit(int L, int R)

{

    int mapa[10];

    for(int i = 0; i < 10; i++)

        mapa[i] = 0;

    for (int i = L; i <= R; i++)

    {

        int copia = i;

        while (copia > 0)

        {

            mapa[copia % 10]++;

            copia /= 10;

        }

    }

    int maxDig = 0, maxFrec = 0;

    for (int i = 0; i < 10; i++)

    {

        if (mapa[i] > maxFrec)

        {

            maxFrec = mapa[i];

            maxDig = i;

        }

    }

    return maxDig;

}

int main(){

    return 0;

}