A function that finds the greatest common divisor of two positive integers can be written through simple recursion and looping. You write the function gcdRecursion to perform the greatest common divisor by recursion and the function gcdIteration to find the greatest common divisor by loop.

**Input:**

Two integers p, q respectively (1 ≤ p,q < 109).

**Output:**

The gcdRecursion and gcdIteration functions return the greatest common divisor of p, q, respectively.

**Template of full code:**

#include<iostream>

#include<string>

#include <string>

#include <sstream>

#include <fstream>

#include <vector>

using namespace std;

/\* END of library \*/

int gcdRecursion(int p, int q)

{

// BEGIN YOUR IMPLEMENTATION [1]

// TODO

// END YOUR EMPLEMENTATION [1]

}

int gcdIteration(int p, int q)

{

hiddenCheck();

// BEGIN YOUR IMPLEMENTATION [2]

// TODO

// END YOUR EMPLEMENTATION [2]

return 0;

}

int main()

{

hiddenCheck();

int p,q;

cin>>p>>q;

cout<<gcdRecursion(p,q)<< " "<<gcdIteration(p,q);

return 0;

}

**For example:**

| **Test** | **Input** | **Result** |
| --- | --- | --- |
| 1 | 5 5 | 5 5 |