Calculates the Least Common Multiple (LCM) of two strictly positive integer numbers.

```
/*
* CVS identifier:
*
* $Id: MathUtil.java,v 1.15 2001/09/14 08:48:51 grosbois Exp $
*
* Class: MathUtil
*
* Description: Utility mathematical methods
*
*
*
* COPYRIGHT:
*
* This software module was originally developed by Raphaël Grosbois and
* Diego Santa Cruz (Swiss Federal Institute of Technology-EPFL); Joel
* Askelöf (Ericsson Radio Systems AB); and Bertrand Berthelot, David
* Bouchard, Félix Henry, Gerard Mozelle and Patrice Onno (Canon Research
* Centre France S.A) in the course of development of the JPEG2000
* standard as specified by ISO/IEC 15444 (JPEG 2000 Standard). This
* software module is an implementation of a part of the JPEG 2000
* Standard. Swiss Federal Institute of Technology-EPFL,
```

Ericsson Radio
* Systems AB and Canon Research Centre France S.A (collectively JJ2000
* Partners) agree not to assert against ISO/IEC and users of the JPEG
* 2000 Standard (Users) any of their rights under the copyright, not
* including other intellectual property rights, for this software module
* with respect to the usage by ISO/IEC and Users of this software module
* or modifications thereof for use in hardware or software products
* claiming conformance to the JPEG 2000 Standard. Those intending to use
* this software module in hardware or software products are advised that
* their use may infringe existing patents. The original developers of
* this software module, JJ2000 Partners and ISO/IEC assume no liability
* for use of this software module or modifications thereof. No license
* or right to this software module is granted for non JPEG 2000 Standard
* conforming products. JJ2000 Partners have full right to use this
* software module for his/her own purpose, assign or donate this
* software module to any third party and to inhibit third parties from
* using this software module for non JPEG 2000 Standard conforming
* products. This copyright notice must be included in all copies or
* derivative works of this software module.
*

```
* Copyright (c) 1999/2000 JJ2000 Partners.
* */
using System;
namespace CSJ2K.j2k.util
{

///

This class contains a collection of utility methods fro mathematical
/// operations. All methods are static.
///
///
public class MathUtil
{

///

Method that calculates the Least Common Multiple (LCM) of two strictly
/// positive integer numbers.
///
///
/// First number
///
///  /// Second number
///
///  public static int lcm(int x1, int x2)
{
if (x1 <= 0 || x2 <= 0) { throw new System.ArgumentException("Cannot compute the least " + "common multiple of two " + "numbers if one, at least," + "is negative."); } int max, min; if (x1 > x2)
{
max = x1;
min = x2;
}
else
{
```

```
max = x2;
min = x1;
}
for (int i = 1; i <= min; i++) { if ((max * i) % min == 0) { return i * max; } } throw new System.ApplicationException("Cannot find the least common multiple of numbers " + x1 + " and " + x2); } ///
Method that calculates the Least Common Multiple (LCM) of several
/// positive integer numbers.
///
///

/// Array containing the numbers.
///
///  public static int lcm(int[] x)
{
if (x.Length < 2) { throw new System.ApplicationException("Do not use this method if there are less than" + " two numbers."); } int tmp = lcm(x[x.Length - 1], x[x.Length - 2]);
for (int i = x.Length - 3; i >= 0; i–)
{
if (x[i] <= 0) { throw new System.ArgumentException("Cannot compute the least " + "common multiple of " + "several numbers where " + "one, at least," + "is negative."); } tmp = lcm(tmp, x[i]); } return tmp; } } } [/csharp]
```