链滴

C# 线性筛法快速求出范围内的所有质数

作者：WayneShao
原文链接：https://ld246.com/article/1540801218073
来源网站：链滴

质数是指在大于1的自然数中，除了1和它本身以外不再有其他因数的数。本文列举了几种求区间内所质数的算法。

<!-- more -->

一般方法

验证质数

我们根据质数的定义可以对某一自然数进行检测，方法如下：

```
/// <summary>
/// 判断自然数是否是质数
/// </summary>
/// <param name="n">需要判断的数</param>
/// <returns>是/否</returns>
public static bool IsPrime(int n)
{
    if (n < 2) return false;
    for (var i = n - 1; i > 1; i--)
    {
        //n除以每个比n小比1大的自然数
        if (n % i == 0)
            //如果有能被整除的，则不是质数
            return false;
    }
    //否则则为质数
    return true;
}
```

获取范围内质数

```
/// <summary>
/// 获取自然数区间内的所有质数
/// </summary>
/// <param name="startInt">自然数区间起始点</param>
/// <param name="endInt">自然数区间终点</param>
/// <returns>自然数区间内的所有质数的集合</returns>
public static IEnumerable<int> GetPrimes(int startInt, int endInt)
{
    for (var i = startInt; i <= endInt; i++)
        if (IsPrime(i))
            yield return i;
}
```

筛法

什么是筛法

科普篇:筛法是一种简单检定质数的算法。据说是古希腊的埃拉托斯特尼（Eratosthenes）发明的，

称埃拉托斯特尼筛法（sieve of Eratosthenes）.

使用筛法求某上限内所有质数

```
/// <summary>
/// 求某上限内的所有质数
/// </summary>
/// <param name="j">上限自然数</param>
/// <returns>上限内所有质数</returns>
public static IEnumerable<int> GenPrime(int j)
{
    var bts = new BitArray(j + 1);
    for (var x = 2; x < bts.Length / 2; x++)
        for (var y = x + 1; y < bts.Length; y++)
            if (bts[y] == false && y % x == 0)
                bts[y] = true;
    for (var x = 2; x < bts.Length; x++)
        if (bts[x] == false)
            yield return x;
}
```

使用筛法求某自然数区间内所有质数

使用两次上面的方法分别求取区间起始点和终点作为上限的所有质数后取差集

```
/// <summary>
/// 获取自然数区间内的所有质数
/// </summary>
/// <param name="startInt">自然数区间起始点</param>
/// <param name="endInt">自然数区间终点</param>
/// <returns>自然数区间内的所有质数的集合</returns>
public static IEnumerable<int> GenPrimes(int startInt, int endInt) =>
    GenPrime(endInt).Except(GenPrime(startInt));
```

快速线性筛法实现

在某位高手的博文中发现了复杂度更低的线性筛法

一般筛法求素数+快速线性筛法求素数

下面是C#的实现

```
/// <summary>
/// 求某上限内的所有质数
/// </summary>
/// <param name="x">上限自然数</param>
/// <returns>上限内所有质数</returns>
private static IEnumerable<int> LinearGenPrime(int x)
{
    var numPrime = 0;
    var ints = new List<int>();
    var isNotPrime = new BitArray(x);
    for (var i = 2; i < x; i++)
```

```
    {
        if (!isNotPrime[i])
        {
            yield return i;
            numPrime++;
        }
        for (var j = 0; j < numPrime && i * ints[j] < x; j++)
        {
            isNotPrime[i * ints[j]] = true;
            if (!Convert.ToBoolean(i % ints[j]))
                break;
        }
    }
}
```

原文链接：C# 线性筛法快速求出范围内的所有质数