**5 kyu**

**Primes in numbers**

4457590% of 764506of 5,303[g964](https://www.codewars.com/users/g964)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/primes-in-numbers/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/primes-in-numbers/solutions/csharp)

[Forks (12)](https://www.codewars.com/kata/primes-in-numbers/forks/csharp)

[Discourse (127)](https://www.codewars.com/kata/primes-in-numbers/discuss/csharp)

* Add to Collection
* |
* Share this kata:

Given a positive number n > 1 find the prime factor decomposition of n. The result will be a string with the following form :

"(p1\*\*n1)(p2\*\*n2)...(pk\*\*nk)"

with the p(i) in increasing order and n(i) empty if n(i) is 1.

Example: n = 86240 should return "(2\*\*5)(5)(7\*\*2)(11)"

<https://www.codewars.com/kata/primes-in-numbers/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

// A function to print all prime

// factors of a given number n

public static Dictionary<int,int> primeFactors(int n)

{

Dictionary<int, int> diccio = new Dictionary<int, int>();

// Print the number of 2s that divide n

while (n % 2 == 0)

{

//Console.Write(2 + " ");

if (diccio.ContainsKey(2))

{

diccio[2]++;

}

else

{

diccio[2] = 1;

}

n /= 2;

}

// n must be odd at this point. So we can

// skip one element (Note i = i +2)

for (int i = 3; i <= Math.Sqrt(n); i += 2)

{

// While i divides n, print i and divide n

while (n % i == 0)

{

if (diccio.ContainsKey(i))

{

diccio[i]++;

}

else

{

diccio[i] = 1;

}

//Console.Write(i + " ");

n /= i;

}

}

// This condition is to handle the case whien

// n is a prime number greater than 2

if (n > 2)

{

// Console.Write(n);

diccio[n] = 1;

}

return diccio;

}

public static String factors(int lst)

{

string ans = "";

foreach(KeyValuePair<int,int> kvp in primeFactors(lst))

{

//Console.WriteLine(kvp.Key + " " + kvp.Value);

ans += "(" + kvp.Key;

if(kvp.Value >1)

{

ans += "\*\*" + kvp.Value + ")";

}else

{

ans += ")";

}

}

return ans ;

}

static void Main(string[] args)

{

int lst = 7775460;

Console.WriteLine( factors(lst));

Console.ReadLine();

}

}

}