Given the first 11 digits of a [GTIN-12](https://en.wikipedia.org/wiki/Global_Trade_Item_Number) barcode as an integer (i.e. without leading zeros), determine the last number by calculating the value of the [check digit](http://www.gs1.org/how-calculate-check-digit-manually).

**Example**

For a = 29104150021, the output should be  
checksum\_part1(a) = 9.

The digit sum can be calculated as follows, using the information in the link above:

You know that you need to multiply the digit at N1 by 3, the digit at N2 by 1, and so on, then add these numbers together: 2 \* 3 + 9 \* 1 + 1 \* 3 + 0 \* 1 + 4 \* 3 + 1 \* 1 + 5 \* 3 + 0 \* 1 + 0 \* 3 + 2 \* 1 + 1 \* 3 = 51

Then, you need to subtract that sum from the nearest equal or higher multiple of ten: 60 - 51 = 9.

**Input/Output**

* **[time limit] 6000ms (cs)**
* **[input] integer64 barcode**

*Constraints:*  
1 ≤ barcode < 1011.

* **[output] integer**

The last digit of the barcode.

<https://codefights.com/challenge/MTgvXpcmjxQzkqpek?utm_source=emailNotification&utm_medium=email&utm_campaign=featuredChallenge>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static int checksum\_part1(long barcode)

{

int sum = 0;

string s = barcode.ToString();

if (barcode.ToString().Length % 2 == 0)

{

for (int i = 0; i < s.Length; i++)

{

if (i % 2 == 0)

{

sum += (int.Parse(s[i].ToString()) \* 3);

}

else

{

sum += (int.Parse(s[i].ToString()) \* 1);

}

}

}

else

{

for (int i = 0; i < s.Length; i++)

{

if (i % 2 == 0)

{

sum += (int.Parse(s[i].ToString()) \* 1);

}

else

{

sum += (int.Parse(s[i].ToString()) \* 3);

}

}

}

int nearestMultiple = sum;

while (nearestMultiple % 10 != 0)

{

nearestMultiple++;

}

Console.WriteLine(sum + " " + nearestMultiple);

return nearestMultiple - sum;

}

static void Main(string[] args)

{

long a = 3600029145;

Console.WriteLine( checksum\_part1(a));

Console.ReadLine();

}

}

}