import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class SINCounter

{

public static void main(String[] args) throws IOException

{

//Variable Declarations and Initializations

String SIN;

int checkDigit, step1, evenDigit = 0, oddDigit = 0, sum = 0;

BufferedReader input = new BufferedReader(new InputStreamReader (System.in));

System.out.println("Social Insurance Number\n======================");

System.out.println("Enter a Social Insurance Number (no spaces): ");

SIN = input.readLine();

//Separate the checkDigit and the rest of the eight digits

checkDigit = Character.getNumericValue(SIN.charAt(8));

SIN = SIN.substring (0,8);

for (int i = 1; i != SIN.length() + 1; i += 2)

{

//Step 1: Multiply the digits in the even positions of the SIN

step1 = Character.getNumericValue(SIN.charAt(i)) \* 2;

//Step 2: Separate the digits individually, this can be done by leaving single digits (already multiplied) and separating the multi-digit numbers (multiplied)

if(step1 >= 10)

{

step1 = (step1 % 10) + Math.round((step1 / 10));

}//end if

evenDigit += step1;

//Step 3: Add together all of the odd position digits that have not been multiplied by 2.

//(which means if the odd number is not the same value as an even number in the SIN then add it to oddDigit)

if (Character.getNumericValue(SIN.charAt(i)) != Character.getNumericValue(SIN.charAt(i - 1)))

{

oddDigit += Character.getNumericValue(SIN.charAt(i - 1));

}//end if

sum = evenDigit + oddDigit;

}//end loop

if (Math.round(((sum / 10) + 0.5)) \* 10 - sum == checkDigit)

{

System.out.println("The check digit of this SIN is correct.");

}//end if

else

{

System.out.println("The check digit of this SIN is NOT correct.");

}//end else

}//end main

}//end class