4DigitCode

package com.company;

import java.util.Arrays;

import java.util.Scanner;

public class Test1 {

public static void main(String args[]) {

int givenNum;

System.out.println("Enter a 4-digit number");

// Create Scanner object

Scanner s = new Scanner(System.in);

// Read the next integer from the screen

givenNum = s.nextInt();

if(givenNum/1000<10&&givenNum/1000>=1) {

RR rr = new RR();

long RC = rr.expCheck(givenNum);

if (RC > 10) {

Sort S = new Sort();

int Wnum = 1;

while (Wnum != 6174) {

int Dnum = S.desending(givenNum);

int Anum = S.asending(Dnum);

int Snum = Dnum - Anum;

System.out.println(Dnum + "-" + Anum + "=" + Snum);

givenNum=Snum;

Wnum=Snum;

}

} else {

System.out.println("This is an Exception");

}

}else{

System.out.println("Not a 4 digit number exit 0");

}

} }

class Sort

{

int desending(int n) {

//Get four digits of given number

int[] digits = new int[4];

for (int i = 0; i < 4; i++) {

digits[i] = n % 10;

n = n / 10;

}

// Get all four dgits in descending order

// in the form of number "desc"

Arrays.sort(digits);

int desc = 0;

for (int i = 3; i >= 0; i--)

desc = desc \* 10 + digits[i];

return desc;

}

int asending(int desendedNum) {

int asendedNum = 0;

while(desendedNum != 0) {

int digit = desendedNum % 10;

asendedNum = asendedNum \* 10 + digit;

desendedNum /= 10;

}

return asendedNum;

}

}

class RR{

long expCheck(long n){

// Store first digits as previous

// digit

long prev\_digit = n % 10;

// Initialize power

long pow = 10;

long res = prev\_digit;

// Iterate through all digits of n,

// note that the digits are

// processed from least significant

// digit to most significant digit.

while (n > 0) {

// Store current digit

long curr\_digit = n % 10;

if (curr\_digit != prev\_digit) {

// Add the current digit to

// the beginning of result

res += curr\_digit \* pow;

// Update previous result

// and power

prev\_digit = curr\_digit;

pow \*= 10;

}

// Remove last digit from n

n = n / 10;

}

return res;

}

}

5DigitCode

import java.util.Arrays;

import java.util.Scanner;

class test2 {

public static void main(String args[]) {

int givenNum;

System.out.println("Enter a 5-digit number");

// Create Scanner object

Scanner s = new Scanner(System.in);

// Read the next integer from the screen

givenNum = s.nextInt();

if(givenNum/10000<10&&givenNum/10000>=1) {

RR rr = new RR();

long RC = rr.expCheck(givenNum);

if (RC > 10) {

Sort S = new Sort();

int Wnum = 1;

while (Wnum != 53955 && Wnum != 74943 && Wnum != 63954 ) {

int Dnum = S.desending(givenNum);

int Anum = S.asending(Dnum);

int Snum = Dnum - Anum;

System.out.println(Dnum + "-" + Anum + "=" + Snum);

givenNum=Snum;

Wnum=Snum;

}

} else {

System.out.println("This is an Exception");

}

}else{

System.out.println("Not a 5 digit number exit 0");

}

} }

class Sort

{

int desending(int n) {

//Get four digits of given number

int[] digits = new int[5];

for (int i = 0; i < 5; i++) {

digits[i] = n % 10;

n = n / 10;

}

// Get all four dgits in descending order

// in the form of number "desc"

Arrays.sort(digits);

int desc = 0;

for (int i = 4; i >= 0; i--)

desc = desc \* 10 + digits[i];

return desc;

}

int asending(int desendedNum) {

int asendedNum = 0;

while(desendedNum != 0) {

int digit = desendedNum % 10;

asendedNum = asendedNum \* 10 + digit;

desendedNum /= 10;

}

return asendedNum;

}

}

class RR{

long expCheck(long n){

// Store first digits as previous

// digit

long prev\_digit = n % 10;

// Initialize power

long pow = 10;

long res = prev\_digit;

// Iterate through all digits of n,

// note that the digits are

// processed from least significant

// digit to most significant digit.

while (n > 0) {

// Store current digit

long curr\_digit = n % 10;

if (curr\_digit != prev\_digit) {

// Add the current digit to

// the beginning of result

res += curr\_digit \* pow;

// Update previous result

// and power

prev\_digit = curr\_digit;

pow \*= 10;

}

// Remove last digit from n

n = n / 10;

}

return res;

}

}

6DigitCode

import java.util.Arrays;

import java.util.Scanner;

class test2 {

public static void main(String args[]) {

int givenNum;

System.out.println("Enter a 6-digit number");

// Create Scanner object

Scanner s = new Scanner(System.in);

// Read the next integer from the screen

givenNum = s.nextInt();

if(givenNum/100000<10&&givenNum/100000>=1) {

RR rr = new RR();

long RC = rr.expCheck(givenNum);

if (RC > 10) {

Sort S = new Sort();

int Wnum = 1;

while (Wnum != 631764 && Wnum != 549945 && Wnum != 851742 ) {

int Dnum = S.desending(givenNum);

int Anum = S.asending(Dnum);

int Snum = Dnum - Anum;

System.out.println(Dnum + "-" + Anum + "=" + Snum);

givenNum=Snum;

Wnum=Snum;

}

} else {

System.out.println("This is an Exception");

}

}else{

System.out.println("Not a 6 digit number exit 0");

}

} }

class Sort

{

int desending(int n) {

//Get four digits of given number

int[] digits = new int[6];

for (int i = 0; i < 6; i++) {

digits[i] = n % 10;

n = n / 10;

}

// Get all four dgits in descending order

// in the form of number "desc"

Arrays.sort(digits);

int desc = 0;

for (int i = 5; i >= 0; i--)

desc = desc \* 10 + digits[i];

return desc;

}

int asending(int desendedNum) {

int asendedNum = 0;

while(desendedNum != 0) {

int digit = desendedNum % 10;

asendedNum = asendedNum \* 10 + digit;

desendedNum /= 10;

}

return asendedNum;

}

}

class RR{

long expCheck(long n){

// Store first digits as previous

// digit

long prev\_digit = n % 10;

// Initialize power

long pow = 10;

long res = prev\_digit;

// Iterate through all digits of n,

// note that the digits are

// processed from least significant

// digit to most significant digit.

while (n > 0) {

// Store current digit

long curr\_digit = n % 10;

if (curr\_digit != prev\_digit) {

// Add the current digit to

// the beginning of result

res += curr\_digit \* pow;

// Update previous result

// and power

prev\_digit = curr\_digit;

pow \*= 10;

}

// Remove last digit from n

n = n / 10;

}

return res;

}

}