# PESEL

1. Napisz metodę isValidPesel(String pesel), która sprawdzi czy podany numer pesel jest poprawny.
2. Napisz metodę getGenderByPesel(String pesel), która zwróci płeć (kobieta/mężczyzna) na podstawie podanego numeru pesel.
3. \*Napisz metodę getDateOfBirth(String pesel), która zwróci datę urodzenia na podstawie podanego numeru pesel. Uwaga na osoby, które się urodziły poza latami 1900–1999!

Dodatkowe materiały:

https://pl.wikipedia.org/wiki/PESEL

# Skoki w dal

Mateusz cały rok przygotowuje się do zawodów sportowych w skoku w dal. Codziennie trenuje, a wynik (długość najdłuższego skoku podaną w centymetrach) zapisuje na kartce. Pozytywną serią treningową Mateusz nazywa każdy ciąg kolejnych dni treningowych, w czasie których uzyskuje on każdego dnia treningu lepszy rezultat niż dnia poprzedniego i którego to ciągu nie można przedłużyć (jest to ostatni zanotowany wynik albo wynik z następnego dnia jest gorszy).

**Przykład**:

478 475 470 480 481 481 475 477 480 482 470

W tym przypadku najdłuższa pozytywna seria treningowa (podkreślona) trwała 4 dni, a Mateusz w jej trakcie poprawił swój wynik o 482-475=7 centymetrów.

1. Napisz funkcję, która zwróci ile pozytywnych serii powyżej 3 dni miał Mateusz
2. Napisz funkcję, która zwróci długość najdłuższej pozytywnej serii oraz o ile centymetrów poprawił w jej trakcie swój wynik
3. \*Napisz funkcję, która zwróci wszystkie pozytywne serie w postaci jej długości oraz o ile centymetrów Mateusz poprawił w jej trakcie swój wynik.

# Liczby pierwsze

Napisz program, która znajdzie liczby pierwsze dla zadanego przedziału [2,n].

Przykład:

Dla n=20 zwróci 2,3,5,7,11,13,17,19

Dodatkowe materiały:

<https://pl.wikipedia.org/wiki/Sito_Eratostenesa>

class TestGC {

public static void main(String[] args) throws InterruptedException {

for (long i=0; i<100000; i++) {

String tmp = new String("test");

Thread.sleep(1000);

}

}

}

Źródła z przykładami przekazywania parametrów do metod

public class Point {

public int x;

public int y;

public Point(int x, int y) {

this.x = x;

this.y = y;

}

@Override

public String toString() {

return "Point{" +

"x=" + x +

", y=" + y +

'}';

}

}

public class SimpleSwap {

public static void main(String[] args) {

int x = 0;

int y = 1;

System.out.println("Values before invoke swap method - x: " + x + " y: " + y);

swap(x, y);

System.out.println("Values after invoke swap method - x: " + x + " y: " + y);

}

private static void swap(int a, int b) {

int tmp;

tmp = a;

a = b;

b = tmp;

}

}

public class SwapFields {

public static void main(String[] args) {

Point point = new Point(0, 1);

System.out.println("Values before invoke swap method - " + point);

swap(point);

System.out.println("Values after invoke swap method - " + point);

System.out.println(point);

}

private static void swap(final Point point) {

int tmp;

tmp = point.x;

point.x = point.y;

point.y = tmp;

}

}

public class SwapObject {

public static void main(String[] args) {

Point point = new Point(0, 1);

System.out.println("Values before invoke swap method - " + point);

swap(point);

System.out.println("Values after invoke swap method - " + point);

}

private static void swap(Point point) {

point = new Point(100, 100);

}

}

|  |
| --- |
| package NowJvc; |
|  |  |
|  | import java.util.ArrayList; |
|  | import java.util.Collections; |
|  | import java.util.List; |
|  |  |
|  | public class User { |
|  | private final String name; |
|  | List<String> friends=new ArrayList<>(); |
|  | public User(String name){ |
|  | this.name=name; |
|  | } |
|  |  |
|  | public void aadFriends(String name){ |
|  | friends.add(name); |
|  | } |
|  |  |
|  | public String getName() { |
|  | return name; |
|  | } |
|  |  |
|  | public List<String> getFriends() { |
|  | // return Collections.unmodifiableList(friends); |
|  | return friends; |
|  | } |
|  | } |

package NowJvc;

import com.sun.java\_cup.internal.runtime.Symbol;

import java.util.List;

import java.util.Scanner;

public class App {

public static void main(String[] args) {

User user1 = new User("Ala");

user1.aadFriends("Krzysiek");

user1.aadFriends("JAnek");

user1.aadFriends("Krzyek");

List<String> friends = user1.getFriends();

System.out.println(friends);

// System.out.println(user1.getFriends());

friends.add("Jola");

System.out.println(user1.getFriends());

//isValidPesel("64091903457");

System.out.println(isValidPesel("64091903457"));

getGenderByPesel("64091903457");

int[] tab1 = new int[]{478, 475, 470, 480, 481, 475, 477, 480, 482, 470};

longJump(tab1);

}

// 1. Napisz metodę isValidPesel(String pesel), która sprawdzi czy podany numer pesel jest poprawny.

public static boolean isValidPesel(String pesel) {

System.out.println("Wpiales 11-to cyfrowy numer pesel "

+ pesel);

// java.util.Scanner scan = new Scanner(System.in);

if (pesel.length() != 11) {

return false;

//break;

}

int[] pes\_int = new int[pesel.length()];

char[] pes\_char = pesel.toCharArray();

for (int i = 0; i < pesel.length(); i++) {

pes\_int[i] = Character.getNumericValue(pes\_char[i]);

}

int b = pes\_int[0] + pes\_int[1] \* 3 + pes\_int[2] \* 7 + pes\_int[3] \* 9 + pes\_int[4] + pes\_int[5] \* 3 +

pes\_int[6] \* 7 + pes\_int[7] \* 9 + pes\_int[8] + pes\_int[9] \* 3 + pes\_int[10];

if (b % 10 == 0) {

return true;

}

//System.out.println("true");

return false;

}

public static void getGenderByPesel(String pesel) {

// if (pesel.length() != 11) {

// return false;

//break;

// }

int[] pes\_int = new int[pesel.length()];

char[] pes\_char = pesel.toCharArray();

for (int i = 0; i < pesel.length(); i++) {

pes\_int[i] = Character.getNumericValue(pes\_char[i]);

}

if (pes\_int[9] % 2 == 0) {

System.out.println("Kobieta");

}

System.out.println("Męższczyzna");

}

public static int longJump(int[] tab) {

for (int i = 0; i < tab.length; i++) {

int d=tab[i + 1];

int e=tab[i + 2];

int f=tab[i + 3];

if ((tab[i] < d) || (d < e) || (e < f)) {

return i;

// System.out.println("znalazłem pozytywną serie z powyżej 3 dni i rozpoczęła się od dnia " +(a-3));

}

int a = i;

System.out.println("znalazłem pozytywną serie z powyżej 3 dni i rozpoczęła się od dnia " + (a - 3));

}

System.out.println("nie znalazłem pozytywnej serii");

}

}