import java.util.ArrayList;

import java.util.LinkedList;

import java.util.stream.Collectors;

import java.util.stream.IntStream;

public class Main {

public static void demoArrayList(){

ArrayList<Integer> list = new ArrayList<>();

list.add(2);

list.add(14);

list.add(20);

list.add(27);

list.add(212);

list.add(7640);

System.out.println("parcours 1");

for (int i = 0; i < list.size(); i++) {

System.out.println(list.get(i));

}

list.removeIf( x -> x%2==1 );

System.out.println("parcours 2");

for (int nombre : list) {

System.out.println(nombre);

}

ArrayList<Integer> list2 =

IntStream.range(0,100)

.mapToObj(x->2\*x)

.map(x->x\*x)

.collect(Collectors.toCollection(ArrayList::new));

for (int i = 0; i < list2.size(); i++) {

System.out.println("list2["+i+"] : "+list2.get(i));

}

System.out.println(list2.stream().filter(x->x==2).findFirst());

for (int i = 0; i < list2.size(); i++) {

if (list2.get(i)==2) System.out.println(list.get(i));

}

}

public static void demoLinkedList(){

LinkedList<String> list = new LinkedList<>();

list.add("AAAA");

list.add("BBBB");

list.add("BBBB");

list.add("Salut");

list.add("BBBB");

list.add("BBBB");

//loza fa O(n^2)

for (int i = 0; i < list.size(); i++) {

System.out.println(list.get(i)); // ! i itérations

}

/// 1 + 2 + 3 + 4 ... + n = n (1 + n) /2 = (n^2)/2 + n/2 = O(n^2)

for (String s:list) { //O(n)

System.out.println(s);

}

/// p:=tete;

/// while (p<>nil) do

/// begin

/// writeLn(p^.val);

/// p:=p^.suiv;

/// end;

}

public static void main(String[] args) {

demoLinkedList();

}

}