package Codes;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.Collections;

public class Optimum

{

int zz;

public Optimum (ArrayList<Double> arraylistone, ArrayList<Double> arraylisttwo, ArrayList<Double> arraylistthree, ArrayList<Double> arraylistfour, ArrayList<Double> arraylistfive, ArrayList<Double> arraylistsix, ArrayList<Double> arraylistseven, ArrayList<Double> arraylisteight, ArrayList<Double> arraylistnine, ArrayList<Double> arraylistten )

{

ArrayList<Double> arraysone=new ArrayList<Double>(Arrays.asList(arraylistone.get(0),arraylisttwo.get(0),arraylistthree.get(0),arraylistfour.get(0),arraylistfive.get(0),arraylistsix.get(0),arraylistseven.get(0),arraylisteight.get(0),arraylistnine.get(0),arraylistten.get(0)));

ArrayList<Double> arraystwo=new ArrayList<Double>(Arrays.asList(arraylistone.get(arraylistone.size()-1),arraylisttwo.get(arraylisttwo.size()-1),arraylistthree.get(arraylistthree.size()-1),arraylistfour.get(arraylistfour.size()-1),arraylistfive.get(arraylistfive.size()-1),arraylistsix.get(arraylistsix.size()-1),arraylistseven.get(arraylistseven.size()-1),arraylisteight.get(arraylisteight.size()-1),arraylistnine.get(arraylistnine.size()-1),arraylistten.get(arraylistten.size()-1)));

//ArrayList<Double> arraycopy=new ArrayList<Double>(Arrays.asList(arraysone));

//ArrayList<Element> arrayList = new ArrayList<Element>(Arrays.asList(array));

ArrayList<Double> arraysonecopy = new ArrayList<Double>();

arraysonecopy.addAll(arraysone); //copy of arraysone

Collections.sort(arraysone);

ArrayList<Double> arraysthree=new ArrayList<Double>();

int a=arraysone.size()-1;

//System.out.println(arraysone);

//System.out.println(arraysonecopy);

//System.out.println(arraystwo);

//System.out.println(a);

//System.out.println(arraysone.get(a));

//System.out.println(arraystwo.indexOf(arraysone.get(a)));

double second=100000;

//double first=0;

//System.out.println(1);

if(arraysone.get(arraysone.size()-1)>20) //if highest ratio is >20

{

while(a>-1 && arraysone.get(a)>20)

{

//int index=arraystwo.indexOf(arraysone.get(a));

double first=(arraysone.get(a)/arraystwo.get(arraysonecopy.indexOf(arraysone.get(a))));

arraysthree.add(first);

//arraysone.set(a,(double)(arraysone.size()-a-1));

//System.out.println(first);

if(first<second)

{

second=first;

}

a--;

}

}

else if(arraysone.get(arraysone.size()-1)>10) //if the highest rantio is between 10 and 20

{

//System.out.println(2);

while(a>-1 && arraysone.get(a)>10)

{

//int index=arraystwo.indexOf(arraysone.get(a));

// System.out.println(3);

double first=(arraysone.get(a)/arraystwo.get(arraysonecopy.indexOf(arraysone.get(a))));

arraysthree.add(first);

// arraysone.set(a,(double)(arraysone.size()-a-1));

//System.out.println(first);

if(first<second)

{

second=first;

}

a--;

}

}

else if(arraysone.get(arraysone.size()-1)>6) //if the highest ratio is between 6 and 10

{

while(a>-1 && arraysone.get(a)>6)

{

//int index=arraystwo.indexOf(arraysone.get(a));

double first=(arraysone.get(a)/arraystwo.get(arraysonecopy.indexOf(arraysone.get(a))));

arraysthree.add(first);

//arraysone.set(a,(double)(arraysone.size()-a-1));

//System.out.println(first);

if(first<second)

{

second=first;

}

a--;

}

}

else

{

//System.out.println("No graph is good"); //none is a good graph

}

//System.out.println(arraysone);

//System.out.println(arraysthree);

//System.out.println(second);

int t=arraysthree.indexOf(second);

int d=arraysone.size()-t-1;

zz=arraysonecopy.indexOf(arraysone.get(d))+1;

//System.out.println(zz); //apply conditions on zz now

//like if zz=0 then return arraylistsix for 1000,1000 like this

}

public int optimumfunction()

{

return zz;

}

public static void main(String[] args)

{

}

}