

|  |
| --- |
| checkingAccount |
| - overDraftLimit: double  + CheckingAccount(id: int, bal: double)  +withdraw (w: double)  + deposit (d: deposit) |
| + toString |

|  |
| --- |
| SavingsAccount |
| -overDraftLimit: double  + CheckingAccount(id: int, bal: double)  +withdraw (w: double)  + deposit (d: deposit) |
| + toString(): String |

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

Account a= new Account(1122,20000.00);

System.out.println(a.toString());

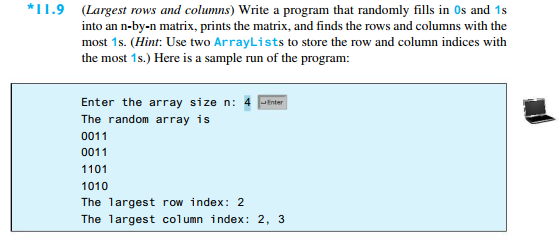
chequingAccount chAccount= new chequingAccount(1122,20000.00);

System.out.println( chAccount.toString() );

savingsAccount svAccount= new savingsAccount(1122,20000.00);

System.out.println( svAccount.toString());

}



public class ArrayCounter {

private int[][] array;

public StackofIntegers RowCountStack;

public StackofIntegers ColCountStack;

private int CurrentMaxCount;

private int CurrentMaxCount2;

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

Scanner input = new Scanner(System.in);

System.out.println("Please enter the size of the array");

int size= input.nextInt();

System.out.println("The Random array is");

ArrayCounter x = new ArrayCounter(size);

x.printArray();

x.maxRow();

x.maxCol();

System.out.print("\nThe largest row index: " );

do{

System.out.print( x.RowCountStack.pop()+" ");

}while(!x.RowCountStack.empty() );

System.out.println();

System.out.print("The largest column index: ");

do{

System.out.print( x.ColCountStack.pop()+" ");

}while(!x.ColCountStack.empty() );

System.out.println();

input.close();

}

public ArrayCounter(int sz) {

this.array= new int[sz][sz];

this.RowCountStack= new StackofIntegers();

this.ColCountStack= new StackofIntegers();

this.fillArray();

}

public void fillArray() {

int temp;

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

for (int j = 0; j < array[i].length; j++) {

temp = (int)(Math.random() \* 2);

array[i][j] = temp;

}

}

}

public void printArray() {

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

System.out.println();

for (int j = 0; j < array.length; j++) {

System.out.print(array[i][j] + " ");

}

}

}

public void maxRow() {

int row= -1; int tempMax=0;

int count=0; //int rowCount= 0;

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

row = i; count = 0; tempMax=0;

for (int j = 0; j < array.length; j++) {

if(array[i][j]==1){

count++;

}

if(count>tempMax){

tempMax= count;

}

}

addRowEntry(RowCountStack, row, tempMax);

}

}

public void addRowEntry(StackofIntegers stack, int row, int tempMax)

{

if(stack.empty()){

stack.push(row+1);

this.CurrentMaxCount= tempMax;

}

else{

if( tempMax > this.CurrentMaxCount ){

do { stack.pop();}

while(!stack.empty());

stack.push(row+1);

this.CurrentMaxCount= tempMax;

}

else{

if( tempMax >= this.CurrentMaxCount ){

stack.push(row+1);

}

}

}

}

public void addColEntry(StackofIntegers stack, int col, int tempMax)

{

if(stack.empty()){

stack.push(col+1);

this.CurrentMaxCount2= tempMax;

}

else{

if( tempMax > this.CurrentMaxCount2 ){

do { stack.pop();}

while(!stack.empty());

stack.push(col+1);

this.CurrentMaxCount2= tempMax;

}

else{

if( tempMax >= this.CurrentMaxCount2 ){

stack.push(col+1);

}

}

}

}

public void maxCol() {

int col= -1; int tempMax=0;

int count=0; int colCount= 0;

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

col = i; count =0;

for (int j = 0; j < array.length; j++) {

if(array[j][i]==1){

count++;

}

if(count>colCount){

tempMax= count;

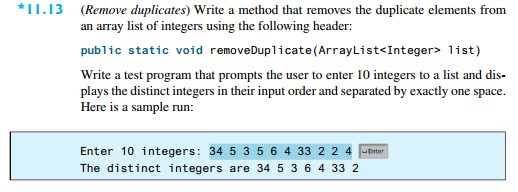
}

}

addColEntry(ColCountStack, col, tempMax);

}

}

}

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

//import java.lang.StringBuilder;

public class StringIntegers {

//public static int[] rawInput;

public static ArrayList<Integer> rawInput;

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

System.out.println("Please Enter 10 inetgers");

Scanner input = new Scanner(System.in);

//rawInput = new int[10];

//strElements = new StringBuilder();

rawInput = new ArrayList <Integer>();

captureInputs(input);

removeDuplicate(rawInput);

printList(rawInput);

input.close();

}

public static void captureInputs(Scanner in){

int i = 0;

while( i < 10){

rawInput.add( in.nextInt() );

i++;

}

}

public static void removeDuplicate ( ArrayList<Integer> list){

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

for (int j = i+1; j < list.size(); j++) {

if( list.get(i) == list.get(j)){

list.remove(j);

}

}

}

}

public static void printList(ArrayList<Integer> list){

System.out.print("\nThe distinct integers are \n");

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

System.out.print(list.get(i)+" ");

}

System.out.println();

}

}