**Clase 3 –Actividad 1 Ejercicio A-**

**Autor: Gil, Matias Ricardo**

1. Utilizando solo tipos primitivos, String (solo usar método length), vectores,

iteraciones simples y condicionales, genere una clase por ejercicio que posea los

siguientes métodos:

b. Dados 3 números y un orden (ascendente o decreciente) que ordene los

mismos y los retorne en un vector de 3

package principal;

import java.util.Scanner;

import java.util.Arrays;

import java.util.Collections;

public class Main {

public static void main(String[] args) {

String orden = "";

System.out.println("Ingrese 3 numeros para llenar el vector");

int numSize = 3;

Integer [] numArray = new Integer [numSize];

//primer bucle for carga el array con numeros ingresados por teclado

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

System.out.print("Ingrese el valor " + (i+1) + "\n");

Scanner numArrayScn = new Scanner(System.in);

int num = numArrayScn.nextInt();

numArray[i] = num;

}

System.out.print("Defina el orden creciente o decreciente para el array. 'C' o 'D' segun corresponda\n");

do {

Scanner ordenScn = new Scanner(System.in);

orden = ordenScn.nextLine().toUpperCase();

if (!(orden.equals("C")) && !(orden.equals("D"))){

System.out.print("Debe ingresar 'C' si desea creciente o 'D' si desea decreciente\n");

}

}

while((!orden.equals("C")) && !(orden.equals("D")));

System.out.print("orden es : \n" + orden);

//Libreria Arrays me permite ordenar creciente. Collections decreciente. Integer para decreciente por que

//solo funciona para arrayas de objetos

if (orden.equals("C")) {

Arrays.sort(numArray);

}else if (orden.equals("D")){

Arrays.sort(numArray, Collections.reverseOrder());

}

System.out.print("\nSu vector es : \n");

//segundo for muestra el array ya cargado

for (int x : numArray){

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

    }

}

}