

Universidad Nacional de Colombia. Sede Bogotá
Facultad de Ingeniería
Departamento de Ing. Industrial y Sistemas
Castro Castro Sergio, Isea Noguera Raúl, Ortiz Ballén Jessica

Códigos utilizados para la implementación de cada uno de los ejercicios (incluyen comentarios):

Ejercicio 1: Escriba un programa que lea 2 números enteros, determine e imprima si son pares, impares, cuál es el mayor y si el primero es múltiplo del segundo.

```
package numeroimparopar;
import java.util.Scanner;
public class NumeroimparOpar {
    public static void main(String[] args) {
        int a,b;
        System.out.println("Ingrese dos numeros y dice cual es par, impar, mayor y si son
multiplos.");
        System.out.println("ingrese el primer numero:");
        Scanner sc = new Scanner(System.in);
        a= sc.nextInt(); //recibe el primer número

        System.out.println("ingrese el segundo numero: ");
        Scanner ss = new Scanner(System.in);
        b=ss.nextInt(); //recibe el segundo número

        if(a%2==0){           //verifica si el primer número es par o impar
            System.out.println("-->El primer numero es par");
        }
        else
        {
            System.out.println("-->El primer numero es impar");
        }

        if(b%2==0){           //verifica si el segundo número es par o impar
            System.out.println("-->El Segundo numero es par");
        }
        else
        {
            System.out.println("-->El segundo numero es impar");
        }

        if(a<b){              //verifica cual de los números es el mayor
            System.out.println("-->El Segundo numero es mayor");
        }
        else
        {
```

```
        System.out.println("-->El primero numero es mayor");
    }

    if(b%a==0){ //verifica si el segundo número es multiplo del primero
        System.out.println("-->El primer numero es multiplo del segundo");
    }
    else
    {
        System.out.println("-->No son multiplos");
    }
}
}
```

Ejercicio 2: Escriba un programa que lea un número enteros, separe el número en sus dígitos, e imprima cada dígito separado uno de otro por 3 espacios. [Ayuda: use el operador módulo; la función potencia en java es Math.pow(double a, double b) y retorna a^b] Por ejemplo si el usuario ingresa 42339 el programa debe imprimir: 4 2 3 3 9

```
import java.util.Scanner;
public class Separar {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Ingresa un numero entero:");
        Integer numeros = scan.nextInt(); // Crea un objeto en base al entero que se le pasa
        char[] nums = numeros.toString().toCharArray(); // Copia los caracteres del objeto
        dentro de una matriz única
        for(char c: nums) {
            System.out.print(c + "   ");
        }
    }
}
```

Ejercicio 3: Escriba un programa que pregunte por números enteros (9999 para salir), calcule e imprima el promedio de los números ingresados. Un ejemplo de entrada es las secuencias [999 1 15 5 9999] indicando el promedio de todos los valores que preceden a 9999 debe ser calculado, este debe imprimir: [The average is: 255].

```
package C_Average;
import java.util.*;
public class Main {
    public static void main(String[] args) {
        try (Scanner sc = new Scanner(System.in)) {
            System.out.println("Ingrese los numeros (Para terminar ingrese el numero 9999):");
        }

        int x;
        int count = 0;
        int n = 0;
        while(sc.hasNext()){ //Nos permite saber si hay mas elementos y movernos al
            siguiente
        }
```

```
x = sc.nextInt();
if(x != 9999){ // Condición para que 9999 haga salir
    n = n + x; // Suma el nuevo valor x con los valores anteriores
    count++; // Cuenta la cantidad de elementos introducidos por el usuario
}else{
    break;
}
}
int average = n/count; // Se define la fórmula del promedio
System.out.println("El promedio es: "+average);
}
}
}
```

Ejercicio 4: Los clientes de un banco quieren realizar retiros de un cajero, Los clientes escriben el monto que quieren retirar, Luego el cajero les devuelve la cantidad solicitada con el menor número de monedas y billetes.

Su tarea es simular el proceso que realiza el banco, dado un monto múltiplo de 50 pesos, usted debe imprimir la cantidad de billetes y monedas que daría de cada denominación.

Para esta tarea se usaran los siguientes tipos de billetes y monedas:

Billetes: 1000, 2000, 5000, 10000, 20000, 50000

Monedas: 50, 100, 200, 500

```
import java.util.*;
```

```
public class Punto4ATM{
```

```
    static private void atmCalc(int n){
```

```
        int Lk,XXk,Xk,Vk,Ilk,lk,D,CC,C,L; //Referencia a los numeros romanos; k refiere a mil, Lk = 50000
```

```
        Lk = n / 50000;           //Este algoritmo, divide el valor por la denominación mas alta, para descubrir cuantas unidades de ese valor se necesitan.
```

```
        n = n - (50000*Lk);
```

```
        XXk = n / 20000;
```

```
        n = n - (20000*XXk);
```

```
        Xk = n / 10000;
```

```
        n = n - (10000*Xk);
```

```
        Vk = n / 5000;
```

```
        n = n - (5000*Vk);
```

```
        Ilk = n / 2000;
```

```
        n = n - (2000*Ilk);
```

```
        lk = n / 1000;
```

```
        n = n - (1000*lk);
```

```
        D = n / 500;
```

```
        n = n - (500*D);
```

```
        CC = n / 200;
```

```
        n = n - (200*CC);
```

```
        C = n / 100;
```

```
n = n - (100*C);
L = n / 50;
n = n - (50*L);
if(Lk != 0)
    System.out.println("Billetes de 50000: " + Lk);
if(XXk != 0)
    System.out.println("Billetes de 20000: " + XXk);
if(Xk != 0)
    System.out.println("Billetes de 10000: " + Xk);
if(Vk != 0)
    System.out.println("Billetes de 5000: " + Vk);
if(IIk != 0)
    System.out.println("Billetes de 2000: " + IIk);
if(Ik != 0)
    System.out.println("Billetes de 1000: " + Ik);
if(D != 0)
    System.out.println("Monedas de 500: " + D);
if(CC != 0)
    System.out.println("Monedas de 200: " + CC);
if(C != 0)
    System.out.println("Monedas de 100: " + C);
if(L != 0)
    System.out.println("Monedas de 50: " + L);

}

public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);
    int x;
    System.out.println("Ingrese el monto a retirar: ");
    while(sc.hasNext()){
        x = sc.nextInt();
        if(x % 50 == 0){           //Simple condicion para evitar que ingresen
            atmCalc(x);           //valores no permitidos
        }else{
            System.out.println("Ingrese un valor multiplo de $50");
        }
    }
    sc.close();
}

}
```

Ejercicio 5: Un número primo es un numero natural que solo tiene 2 divisores, 1 y sí mismo. Dado un número entero (máximo 10^9) imprima una lista ordenada de los números primos menores o iguales al número ingresado.

```
import java.util.Scanner;

public class Punto5ListaPrimos {

    private static void primeList(int n){

        for (int i = 2; i <= n; i++) {

            boolean isPrime = true;
            for (int j = 2; j < i; j++) { //Algoritmo que identifica que numero es primo
                if(i % j == 0){ //desde 2 hasta el valor ingresado
                    isPrime = false;
                    break;
                }
            }
            if(isPrime)
                System.out.println(i + " ");
        }

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        int x;
        System.out.println("Ingresa el numero: ");
        while(sc.hasNextLine()){
            x = sc.nextInt();
            primeList(x);
        }

        sc.close();
    }

}
```

Ejercicio 6: Realizar un programa que nos permita jugar el clásico juego de Tic-Tac-Toe, los jugadores toman turnos para marcar 'O' o 'X' en una de las celdas disponibles de un tablero de 3x3. Un jugador ingrese la coordenada x,y. y esta celda debe ser marcada con la ficha correspondiente del jugador. Cuando el juego se termina se debe imprimir un mensaje anunciando el ganador si lo hay, en caso contrario... ¡ambos pierden!.

```
package tictactoedos;

import java.util.Scanner;

public class Tictactoedos {
```

Programación Orientada a Objetos. Taller 1. Septiembre 2015.

```
private int c;

private char p[]=new char[10];

private char jugador;

public static void main(String[] args) {

    String f;

    Tictactoeodos Toe=new Tictactoeodos();

    do{

        Toe.newBoard();

        Toe.play();

        System.out.println ("Volvemos a Jugar ok?");

        Scanner in =new Scanner(System.in);

        f=in.nextLine();

    }while (f.equals("ok"));           //confirma si el jugador quiere volver a jugar

    String g;

    Tictactoeodos jugar=new Tictactoeodos();

}

public void newBoard()

{

    char matriz[] = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9'};

    int i;

    c = 0;

    jugador = 'X';

    for (i=1; i<10; i++) p[i]=matriz[i];

    currentBoard();

}

public String currentBoard()

{
```

Programación Orientada a Objetos. Taller 1. Septiembre 2015.

```
System.out.println( "\t" + p[1] + " | " + p[2] + " | " + p[3]);
System.out.println( "\t ___/___/___ ");
System.out.println( "\t" + p[4] + " | " + p[5] + " | " + p[6]);
System.out.println( "\t ___/___/___ ");
System.out.println( "\t" + p[7] + " | " + p[8] + " | " + p[9]);
System.out.println( "\t | | ");
System.out.println( "\n\n");
return "currentBoard";
}

public void play()
{
    int poner;
    char blank = ' ';

    System.out.println( "Jugador " + getPlayer() + " empezará primero con la 'X'");

    do {
        currentBoard();          // muestra el tablero

        System.out.println( "\n Jugador " + getPlayer() + " escoja la posición");

        boolean posc = true;
        while (posc) {

            Scanner in = new Scanner (System.in);
            poner=in.nextInt();
            posc = checkPosn(poner); //verifica la posicion ingresada por el jugador
            if(posc==false)
```

Programación Orientada a Objetos. Taller 1. Septiembre 2015.

```
p[poner]=getPlayer();

}

System.out.println( "Bien hecho" );

currentBoard();           // muestra el nuevo tablero

nextPlayer();

}while ( checkWinner() == blank ); //sigue preguntando mientras aun halla posiciones
libres en el tablero

}

public char checkWinner() //metodo que verifica quien de los jugadores es el ganador
{
    char Winner = ' ';

    // Check if X wins
    if (p[1] == 'X' && p[2] == 'X' && p[3] == 'X') Winner = 'X';
    if (p[4] == 'X' && p[5] == 'X' && p[6] == 'X') Winner = 'X';
    if (p[7] == 'X' && p[8] == 'X' && p[9] == 'X') Winner = 'X';
    if (p[1] == 'X' && p[4] == 'X' && p[7] == 'X') Winner = 'X';
    if (p[2] == 'X' && p[5] == 'X' && p[8] == 'X') Winner = 'X';
    if (p[3] == 'X' && p[6] == 'X' && p[9] == 'X') Winner = 'X';
    if (p[1] == 'X' && p[5] == 'X' && p[9] == 'X') Winner = 'X';
    if (p[3] == 'X' && p[5] == 'X' && p[7] == 'X') Winner = 'X';
    if (Winner == 'X' )
    {System.out.println("El jugador 1 tiene Triqui" );
```



```
        return Winner;
    }

    // Check if O wins
    if (p[1] == 'O' && p[2] == 'O' && p[3] == 'O') Winner = 'O';
    if (p[4] == 'O' && p[5] == 'O' && p[6] == 'O') Winner = 'O';
    if (p[7] == 'O' && p[8] == 'O' && p[9] == 'O') Winner = 'O';
    if (p[1] == 'O' && p[4] == 'O' && p[7] == 'O') Winner = 'O';
    if (p[2] == 'O' && p[5] == 'O' && p[8] == 'O') Winner = 'O';
    if (p[3] == 'O' && p[6] == 'O' && p[9] == 'O') Winner = 'O';
    if (p[1] == 'O' && p[5] == 'O' && p[9] == 'O') Winner = 'O';
    if (p[3] == 'O' && p[5] == 'O' && p[7] == 'O') Winner = 'O';
    if (Winner == 'O' )
    {
        System.out.println( "EL jugador 2 tiene Triqui" );
        return Winner; }

    // verifica si no hay ganadores
    for(int i=1;i<10;i++)
    {
        if(p[i]=='X' || p[i]=='O')
        {
            if(i==9)
            {
                char Draw='D';

                System.out.println(" Nadie Gana ");

                return Draw;
            }
        }
    }

    else
```

```
break;
```

```
}
```

```
return Winner;
```

```
}
```

```
public boolean checkPosn(int poner) //metodo que indica o pregunta al jugador la  
posicion que va a ingresar
```

```
{
```

```
    if (p[poner] == 'X' || p[poner] == 'O')
```

```
    {
```

```
        System.out.println("ya fue tomada, escoja otra posición");
```

```
        return true;
```

```
    }
```

```
    else {
```

```
        return false;
```

```
    }
```

```
}
```

```
public void nextPlayer() //metodo que solicita al siguiente jugador
```

```
{
```

```
    if (jugador == 'X')
```

```
        jugador = 'O';
```

```
    else jugador = 'X';
```

```
}
```

```
public char getPlayer()    //metodo que guarda la accion del jugador en curso  
{  
    return jugador;  
}  
}
```