

Taller 12

Respuestas

URL de la carpeta "Taller12" de su repositorio GitHub

https://github.com/JuanT22/RepoIP2024_TACURI_JUAN/tree/main/Taller12

A continuación, copie el código fuente Java de los programas desarrollados

Problema 1: Contar pares en vector

```
package paresenvector1;
import java.util.Scanner;

public class ParesEnVector1 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Leer el tamaño del vector
        System.out.print("Ingrese el tamaño del vector (N): ");
        int n = sc.nextInt();

        // Leer los elementos del vector
        int[] vector = new int[n];
        System.out.print("Ingrese los elementos del vector separados por espacios "
            + "o uno por uno: ");
        for (int i = 0; i < n; i++) {
            vector[i] = sc.nextInt();
        }

        // Contar pares y mostrar resultado
        int cantidadPares = contarPares(vector);
        System.out.println(cantidadPares + " elementos del vector son números
pares.");
    }

    // Función que cuenta los números pares en el vector
    public static int contarPares(int[] vector) {
        int contador = 0;
        for (int numero : vector) {
            if (numero % 2 == 0) {
                contador++;
            }
        }
        return contador;
    }
}
```

Problema 2: Sumar los dígitos de un número

```
package sumadigitos1;
import java.util.Scanner;

public class SumaDigitos1 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int numero,suma;

        System.out.print("Ingrese un número entero: ");
```

```

        numero = scanner.nextInt();
        suma = sumaDigitos(numero);
        System.out.println("La suma de los digitos de " + numero + " es: " + suma);
    }

    public static int sumaDigitos(int numero) {
        if (numero < 10) {
            return numero;
        } else {
            return numero % 10 + sumaDigitos(numero / 10);
        }
    }
}

```

Problema 3: Máximo Común Divisor – Método Euclides (Solución iterativa)

```

package maximocomundivisor1;
import java.util.Scanner;

public class MaximoComunDivisor1 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int m,n;

        System.out.print("Ingrese el primer número (m): ");
        m = sc.nextInt();

        System.out.print("Ingrese el segundo número (n): ");
        n = sc.nextInt();

        int resultado = mcd(m, n);
        System.out.println("El MCD de " + m + " y " + n + " es: " + resultado);
    }

    public static int mcd(int m, int n) {
        while (n != 0) {
            int r = m % n;
            m = n;
            n = r;
        }
        return m;
    }
}

```

Problema 4: Máximo Común Divisor – Método Euclides (Solución recursiva)

```

package maximocomundivisor2;
import java.util.Scanner;

public class MaximoComunDivisor2 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int m,n;

        System.out.print("Ingrese el primer número (m): ");
        m = sc.nextInt();

        System.out.print("Ingrese el segundo número (n): ");
        n = sc.nextInt();

        int resultado = mcd(m, n);
        System.out.println("El MCD de " + m + " y " + n + " es: " + resultado);
    }
}

```

```

    }

    public static int mcd(int m, int n) {
        if (n == 0) {
            return m;
        } else {
            return mcd(n, m % n);
        }
    }
}

```

Problema 5: Máximo común divisor (mejorado)

```

package maximocomundivisor3;
import java.util.InputMismatchException;
import java.util.Scanner;

public class MaximoComunDivisor3 {

    public static void main(String[] args) {
        int m, n, resultado;
        m = leerEnteroPositivo("Ingrese el primer número (m): ");
        n = leerEnteroPositivo("Ingrese el segundo número (n): ");

        resultado = mcd(m, n);
        System.out.println("El MCD de " + m + " y " + n + " es: " + resultado);
    }

    public static int leerEnteroPositivo(String mensaje) {
        Scanner sc = new Scanner(System.in);
        int numero = 0;
        boolean entradaValida = false;

        while (!entradaValida) {
            try {
                System.out.print(mensaje);
                numero = sc.nextInt();
                if (numero > 0) {
                    entradaValida = true;
                } else {
                    System.err.println("Error: El número debe ser positivo.");
                }
            } catch (InputMismatchException e) {
                System.err.println("Error: Ingrese un número entero válido.");
                sc.next();
            }
        }
        return numero;
    }

    public static int mcd(int m, int n) {
        if (n == 0) {
            return m;
        } else {
            return mcd(n, m % n);
        }
    }
}

```

Problema 6: País más densamente poblado

```

package paises1;
import java.io.BufferedReader;

```

```

import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

public class Países1 {
    public static void main(String[] args) {
        String nombreArchivo = "Países.csv";
        double maxDensidad = -1;
        String nombrePaís = "";
        String códigoPaís = "";

        try (BufferedReader br = new BufferedReader(new
FileReader("data/Países.csv"))) {
            br.readLine(); // Saltar encabezado

            String línea;
            int numLínea = 1;

            while ((línea = br.readLine()) != null) {
                numLínea++;
                línea = línea.trim();
                if (línea.isEmpty()) continue;

                String[] campos = línea.split(":");
                if (campos.length < 4) {
                    System.err.println("Error en línea " + numLínea + ": Formato
incorrecto");
                    continue;
                }

                try {
                    String nombre = campos[0];
                    String código = campos[1];
                    int población = Integer.parseInt(campos[2]);
                    double superficie = Double.parseDouble(campos[3]);

                    if (superficie <= 1000) continue;

                    double densidad = población / superficie;

                    if (densidad > maxDensidad) {
                        maxDensidad = densidad;
                        nombrePaís = nombre;
                        códigoPaís = código;
                    }

                } catch (NumberFormatException e) {
                    System.err.println("Error en línea " + numLínea + ": Valores
numéricos inválidos");
                }
            }

            } catch (FileNotFoundException e) {
                System.err.println("Error: Archivo no encontrado - " + nombreArchivo);
            } catch (IOException e) {
                System.err.println("Error de lectura: " + e.getMessage());
            }

            if (maxDensidad != -1) {
                System.out.println("Nombre: " + nombrePaís);
                System.out.println("Código: " + códigoPaís);
                System.out.printf("Densidad poblacional: %.2f hab/km²%n", maxDensidad);
            } else {
                System.out.println("No se encontraron países que cumplan los
criterios.");
            }
        }
    }
}

```

```
}  
}
```

Problema 7: Edad promedio de mujeres

```
package personas5;  
  
import java.io.*;  
import java.util.*;  
  
public class Personas5 {  
  
    public static class EdadNegativaException extends Exception {  
        public EdadNegativaException(String message) {  
            super(message);  
        }  
    }  
  
    public static void main(String[] args) {  
        try {  
            File archivo = new File("data/Personas.txt");  
            Scanner f = new Scanner(archivo);  
            String[] persona;  
            ArrayList<String> nombres = new ArrayList();  
            ArrayList<String> generos = new ArrayList();  
            ArrayList<Integer> edades = new ArrayList();  
  
            while (f.hasNextLine()) {  
                persona = f.nextLine().split(",");  
                nombres.add(persona[0]);  
                generos.add(persona[1]);  
  
                try {  
                    int edad = Integer.parseInt(persona[2]);  
                    if (edad < 0) {  
                        throw new EdadNegativaException("Edad negativa encontrada: " + edad);  
                    }  
                    edades.add(edad);  
                } catch (NumberFormatException e) {  
                    System.err.println("Error al convertir la edad: " + e.getMessage());  
                }  
            }  
            f.close();  
  
            int suma = 0, cf = 0;  
            for (int i = 0; i <= edades.size() - 1; i++) {  
                if (generos.get(i).equals("F")) {  
                    suma += edades.get(i);  
                    cf++;  
                }  
            }  
            double media = (double) suma / cf;  
            System.out.println("Promedio de edad de mujeres: " + media);  
  
        } catch (FileNotFoundException e) {  
            System.err.println("ERROR: Archivo no encontrado:\n" + e.getMessage());  
        } catch (EdadNegativaException e) {  
            System.err.println("ERROR: " + e.getMessage());  
        } catch (Exception e) {  
            System.err.println("ERROR general:\n" + e.getMessage());  
        }  
    }  
}
```

}

.-