Problema 1.-

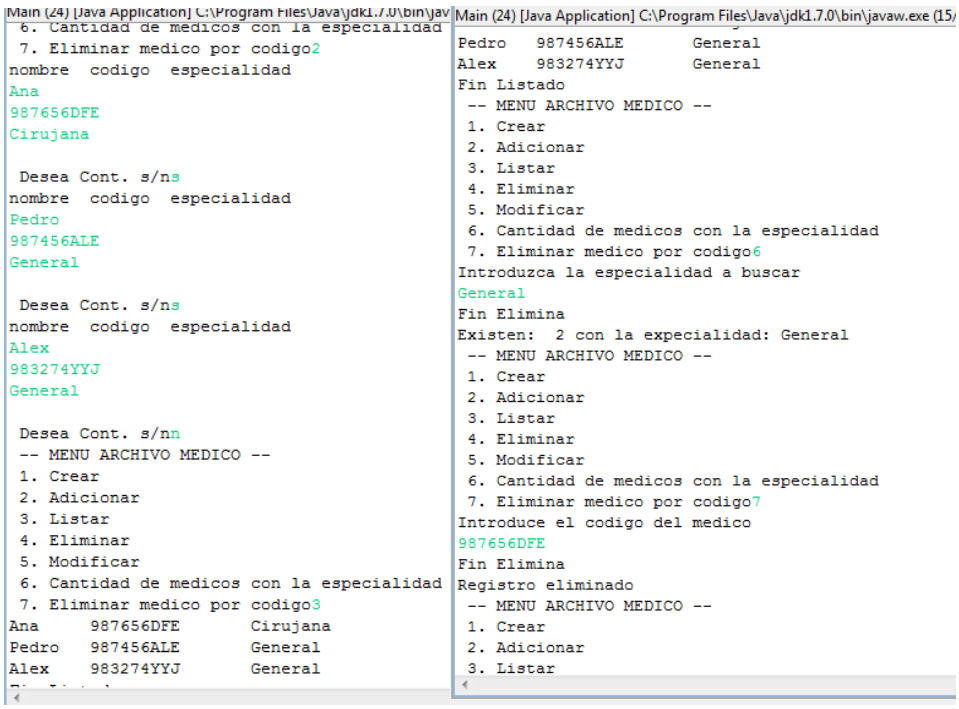
```
public class Main {

	/**
	 * @author Alexander Humberto Nina Pacajes
	 * 1. -Tenemos el siguiente registro de médicos que trabajan en un hospital:
	 * a) Mostrar la cantidad de médicos por especialidad
	 * b) Mostrar a los médicos que tienen más de 2 especialidades
	 * c) Eliminar al médico con codMedico 987656DFE
	 */
	public static void main(String[] args) throws ClassNotFoundException, IOException {
		// TODO Auto-generated method stub
		ArchMedico m=new ArchMedico("Medico.dat");
		Scanner lee=new Scanner(System.in);
		int op;
		do{
			System.out.print(" -- MENU ARCHIVO MEDICO --");
			System.out.print("\n 1. Crear" +
					"\n 2. Adicionar" +
					"\n 3. Listar" +
					"\n 4. Eliminar" +
					"\n 5. Modificar" +
					"\n 6. Cantidad de medicos con la especialidad"+
					"\n 7. Eliminar medico por codigo");

			op=lee.nextInt();
			switch(op){
			case 1:	m.crear();
				break;
			case 2:	m.adicionar();
				break;
			case 3: m.listar();
				break;
			case 4: System.out.println("Inttoducir el codigo del Medico");
				if(m.Eliminar(lee.next()))System.out.println("Registro
Eliminado");
				else System.out.println("Registro no eliminado");
				break;
			case 5: m.Modificar();
				break;
			case 6: m.cantMedicosPorEspecialidad();
				break;
			case 7: System.out.println("Introduce el codigo del medico");
				if(m.eliminarCodMed(lee.next()))System.out.println("Registro
eliminado");
				else System.out.println("No existe ese medico");
				break;
			}
import java.util.*;
public class RegMedico implements java.io.Serializable{
	private String nombre;
	private String codigo;
	private String especialidad;
	public String getNombre() {
		return nombre;
	}
	public void setNombre(String nombre) {
		this.nombre = nombre;
	}
	public String getCodigo() {
		return codigo;
	}
```

```
	public void setCodigo(String codigo) {
		this.codigo = codigo;
	}
	public String getEspecialidad() {
		return especialidad;
	}
	public void setEspecialidad(String especialidad) {
		this.especialidad = especialidad;
	}
	public void leer(){
		Scanner lee=new Scanner(System.in);
		System.out.println("nombre\tcodigo\tespecialidad");
		this.nombre=lee.next();
		this.codigo=lee.next();
		this.especialidad=lee.next();
	}
	public void mostrar(){
		System.out.println(this.nombre+"\t"+this.codigo+"\t"+this.especialidad);
	}
}
public class ArchMedico {
	private String nomArch;
	private RegMedico rMed;
	public ArchMedico(String nomArch) {
		this.nomArch = nomArch;
	}
	public void crear()throws ClassNotFoundException, IOException{
		ObjectOutputStream archMed = new ObjectOutputStream(new
FileOutputStream(nomArch));
		archMed.close();
	}
	public void adicionar()throws ClassNotFoundException, IOException{
		String op;
		Scanner lee = new Scanner(System.in);
		ObjectOutputStream archMed = null;
		try{
			if(new File(nomArch).exists())
				archMed = new AddObjectOutputStream(new
FileOutputStream(nomArch,true));
			else
				archMed = new ObjectOutputStream(new
FileOutputStream(nomArch,true));
			do{
				rMed = new RegMedico();
				rMed.leer();
				archMed.writeObject(rMed);
				System.out.print("\n Desea Cont. s/n");
				op = lee.next();
			}while(op.equals("s"));
			archMed.close();
		}catch(Exception e){
			System.out.print("\n FIN ADICIONA");
		}
	}
	public void listar() throws ClassNotFoundException, IOException{
		ObjectInputStream archMed =null;
		try {
			archMed=new ObjectInputStream(new FileInputStream(nomArch));
			while(true){
				rMed=new RegMedico();
				rMed = (RegMedico)archMed.readObject();
				rMed.mostrar();
			}
```

```
		} catch (Exception e) {
			System.out.println("Fin Listado");
		}finally{
			archMed.close();
		}
	}
	public boolean Eliminar(String codMed)throws ClassNotFoundException,IOException{
		boolean sw = false;
		ObjectInputStream archMed = null;
		ObjectOutputStream aCopia = null;
		try {
			archMed = new ObjectInputStream(new FileInputStream(nomArch));
			aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
			while(true){
				rMed = new RegMedico();
				rMed = (RegMedico)archMed.readObject();
				if(rMed.getCodigo().equals(codMed))
					sw = true;
				else
					aCopia.writeObject(rMed);
			}
		} catch (Exception e) {
			System.out.println("Fin Elimina");
		}finally{
			archMed.close();
			aCopia.close();
			File f1 = new File(nomArch);
			File f2 = new File("copia.dat");
			f1.delete();
			f2.renameTo(f1);
		}
		return sw;
	}
public void cantMedicosPorEspecialidad()throws ClassNotFoundException,IOException{
		System.out.println("Introduzca la especialidad a buscar");
		Scanner lee=new Scanner(System.in);
		String esp=lee.next();
		int n=0;
		ObjectInputStream archMed = null;
		ObjectOutputStream aCopia = null;
		try {
			archMed = new ObjectInputStream(new FileInputStream(nomArch));
			aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
			while(true){
				rMed = new RegMedico();
				rMed = (RegMedico)archMed.readObject();
				if(rMed.getEspecialidad().equals(esp))
					n++;
				else
					aCopia.writeObject(rMed);
			}
		} catch (Exception e) {
			System.out.println("Fin Elimina");
		}finally{
			archMed.close();
			aCopia.close();
			File f1 = new File(nomArch);
			File f2 = new File("copia.dat");
			f1.delete();
			f2.renameTo(f1);
		}
		System.out.println("Existen:  "+n+" con la expecialidad: "+esp);
	}
```

```
	public boolean eliminarCodMed(String codMed)throws
ClassNotFoundException,IOException{
		boolean sw = false;
		ObjectInputStream archMed = null;
		ObjectOutputStream aCopia = null;
		try {
			archMed = new ObjectInputStream(new FileInputStream(nomArch));
			aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
			while(true){
				rMed = new RegMedico();
				rMed = (RegMedico)archMed.readObject();
				if(rMed.getCodigo().equals(codMed))
					sw = true;
				else
					aCopia.writeObject(rMed);
			}
		} catch (Exception e) {
			System.out.println("Fin Elimina");
		}finally{
			archMed.close();
			aCopia.close();
			File f1 = new File(nomArch);
			File f2 = new File("copia.dat");
			f1.delete();
			f2.renameTo(f1);
		}
		return sw;
```

Consola.-



Problema 2.-

```
import java.util.Scanner;

public class RegCliente implements java.io.Serializable {
	private String codigo;
	private String na;
	private int edad;

	public int getEdad() {
```

```
            return edad;
      }

      public void setEdad(int edad) {
            this.edad = edad;
      }

      public String getCodigo() {
            return codigo;
      }

      public void setCodigo(String codigo) {
            this.codigo = codigo;
      }

      public String getNa() {
            return na;
      }

      public void setNa(String na) {
            this.na = na;
      }
      public void mostrar() {
            System.out.println("Nombre: " + this.na);
            System.out.println("Codigo: " + this.codigo);
            System.out.println("Edad: " + this.edad);
      }
      public void leer() {
            Scanner lee = new Scanner(System.in);
            System.out.println("Introduce el cod nom edad");
            this.codigo = lee.next();
            this.na = lee.next();
            this.edad = lee.nextInt();
      }
}
public class ArchivoCliente {
      private String nomArch;
      private RegCliente rClien;
      public ArchivoCliente(String nomArch) {
            this.nomArch = nomArch;
      }
      public void crear()throws ClassNotFoundException, IOException{
            ObjectOutputStream archMed = new ObjectOutputStream(new
FileOutputStream(nomArch));
            archMed.close();
      }
      public void adicionar()throws ClassNotFoundException, IOException{
            String op;
            Scanner lee = new Scanner(System.in);
            ObjectOutputStream archClien = null;
            try{
                  if(new File(nomArch).exists())
                        archClien = new AddObjectOutputStream(new
FileOutputStream(nomArch,true));
                  else
                        archClien = new ObjectOutputStream(new
FileOutputStream(nomArch,true));
                  do{
                        rClien = new RegCliente();
                        rClien.leer();
                        archClien.writeObject(rClien);
                        System.out.print("Desea Cont. s/n");
                        op = lee.next();
```

```
            }while(op.equals("s"));
            archClien.close();
      }catch(Exception e){
            System.out.print("\n FIN ADICIONA");
      }
}
public void listar() throws ClassNotFoundException, IOException{
      ObjectInputStream archClien =null;
      try {
            archClien=new ObjectInputStream(new FileInputStream(nomArch));
            while(true){
                  rClien=new RegCliente();
                  rClien = (RegCliente)archClien.readObject();
                  rClien.mostrar();
            }
      } catch (Exception e) {
            System.out.println("Fin Listado");
      }finally{
            archClien.close();
      }
}
public boolean Eliminar(String codMed)throws ClassNotFoundException,IOException{
      boolean sw = false;
      ObjectInputStream archClien = null;
      ObjectOutputStream aCopia = null;
      try {
            archClien = new ObjectInputStream(new FileInputStream(nomArch));
            aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
            while(true){
                  rClien = new RegCliente();
                  rClien = (RegCliente)archClien.readObject();
                  if(rClien.getCodigo().equals(codMed))
                        sw = true;
                  else
                        aCopia.writeObject(rClien);
            }
      } catch (Exception e) {
            System.out.println("Fin Elimina");
      }finally{
            archClien.close();
            aCopia.close();
            File f1 = new File(nomArch);
            File f2 = new File("copia.dat");
            f1.delete();
            f2.renameTo(f1);
      }
      return sw;
}

public void Modificar()throws ClassNotFoundException,IOException{
      Scanner lee = new Scanner(System.in);
      System.out.println("Intr. cod a modificar");
      String op, codMed = lee.next();
      ObjectInputStream archClien = null;
      ObjectOutputStream aCopia = null;
      try {
            archClien = new ObjectInputStream(new FileInputStream(nomArch));
            aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
            while(true){
                  rClien = new RegCliente();
                  rClien = (RegCliente)archClien.readObject();
                  if(rClien.getCodigo().equals(codMed)){
```

```
                              rClien.mostrar();   System.out.println("Desea modificar
s/n");
                              op = lee.next();
                              if(op.equals("s")){
                                    rClien.leer();
                                    rClien.mostrar();
                              }
                        }
                        aCopia.writeObject(rClien);
                  }
            } catch (Exception e) {
                  System.out.println("Fin modifica");
            }finally{
                  archClien.close();   aCopia.close();
                  File f1 = new File(nomArch);       File f2 = new File("copia.dat");
                  f1.delete();   f2.renameTo(f1);
            }
      }
}
import java.util.Scanner;


public class RegSolicitud implements java.io.Serializable{
      private String idSolicitud;
      private String []libros=new String [10];
      private String idCliente;
      public String getIdSolicitud() {
            return idSolicitud;
      }
      public void setIdSolicitud(String idSolicitud) {
            this.idSolicitud = idSolicitud;
      }
      public String[] getLibros() {
            return libros;
      }
      public void setLibros(String[] libros) {
            this.libros = libros;
      }
      public String getIdCliente() {
            return idCliente;
      }
      public void setIdCliente(String idCliente) {
            this.idCliente = idCliente;
      }
      public void mostrar(){
            System.out.println("id solicitud: "+this.idSolicitud);
            System.out.println("libros");
            for (int i = 0; i < this.libros.length; i++) {
                  System.out.println("Libro"+(i+1)+" : "+this.libros[i]);
            }
            System.out.println("id Cliente; "+this.idCliente);
      }
      public void leer(){
            Scanner lee=new Scanner(System.in);
            System.out.println("introduzca datos");
            this.idSolicitud=lee.next();
            int a=lee.nextInt();
            for (int i = 0; i < a; i++) {
                  this.libros[i]=lee.next();
            }
            this.idCliente=lee.next();
      }
}
```

```
public class ArchivoSolicitud {
	private String nomArch;
	private RegSolicitud rSol;
	public ArchivoSolicitud(String nomArch) {
		this.nomArch = nomArch;
	}
	public void crear()throws ClassNotFoundException, IOException{
		ObjectOutputStream archSol = new ObjectOutputStream(new
FileOutputStream(nomArch));
		archSol.close();
	}
	public void adicionar()throws ClassNotFoundException, IOException{
		String op;
		Scanner lee = new Scanner(System.in);
		ObjectOutputStream archSol = null;
		try{
			if(new File(nomArch).exists())
				archSol = new AddObjectOutputStream(new
FileOutputStream(nomArch,true));
			else
				archSol = new ObjectOutputStream(new
FileOutputStream(nomArch,true));
			do{
				rSol = new RegSolicitud();
				rSol.leer();
				archSol.writeObject(rSol);
				System.out.print("Desea Cont. s/n");
				op = lee.next();
			}while(op.equals("s"));
			archSol.close();
		}catch(Exception e){
			System.out.print("\n FIN ADICIONA");
		}
	}
	public void listar() throws ClassNotFoundException, IOException{
		ObjectInputStream archSol =null;
		try {
			archSol=new ObjectInputStream(new FileInputStream(nomArch));
			while(true){
				rSol=new RegSolicitud();
				rSol = (RegSolicitud)archSol.readObject();
				rSol.mostrar();
			}
		} catch (Exception e) {
			System.out.println("Fin Listado");
		}finally{
			archSol.close();
		}
	}
	public boolean Eliminar(String codMed)throws ClassNotFoundException,IOException{
		boolean sw = false;
		ObjectInputStream archSol = null;
		ObjectOutputStream aCopia = null;
		try {
			archSol = new ObjectInputStream(new FileInputStream(nomArch));
			aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
			while(true){
				rSol = new RegSolicitud();
				rSol = (RegSolicitud)archSol.readObject();
				if(rSol.getIdSolicitud().equals(codMed))
					sw = true;
				else
					aCopia.writeObject(rSol);
```

```
                }
            } catch (Exception e) {
                System.out.println("Fin Elimina");
            }finally{
                archSol.close();
                aCopia.close();
                File f1 = new File(nomArch);
                File f2 = new File("copia.dat");
                f1.delete();
                f2.renameTo(f1);
            }
            return sw;
        }

        public void Modificar()throws ClassNotFoundException,IOException{
            Scanner lee = new Scanner(System.in);
            System.out.println("Intr. cod a modificar");
            String op, codMed = lee.next();
            ObjectInputStream archSol = null;
            ObjectOutputStream aCopia = null;
            try {
                archSol = new ObjectInputStream(new FileInputStream(nomArch));
                aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
                while(true){
                    rSol = new RegSolicitud();
                    rSol = (RegSolicitud)archSol.readObject();
                    if(rSol.getIdSolicitud().equals(codMed)){
                        rSol.mostrar();   System.out.println("Desea modificar s/n");
                        op = lee.next();
                        if(op.equals("s")){
                            rSol.leer();
                            rSol.mostrar();
                        }
                    }
                    aCopia.writeObject(rSol);
                }
            } catch (Exception e) {
                System.out.println("Fin modifica");
            }finally{
                archSol.close();   aCopia.close();
                File f1 = new File(nomArch);        File f2 = new File("copia.dat");
                f1.delete();   f2.renameTo(f1);
            }
        }

}
import java.util.Scanner;


public class RegLibro implements java.io.Serializable{
        private String idLibro;
        private String titulo;
        private String autor;
        public String getIdLibro() {
            return idLibro;
        }
        public void setIdLibro(String idLibro) {
            this.idLibro = idLibro;
        }
        public String getTitulo() {
            return titulo;
        }
        public void setTitulo(String titulo) {
```

```
            this.titulo = titulo;
      }
      public String getAutor() {
            return autor;
      }
      public void setAutor(String autor) {
            this.autor = autor;
      }
      public void mostrar(){
            System.out.println("id libro: "+this.idLibro);
            System.out.println("titulo.: "+this.titulo);
            System.out.println("Autor: "+this.autor);
      }
      public void leer(){
            Scanner lee=new Scanner(System.in);
            System.out.println("Introduzca id titulo autor");
            this.idLibro=lee.next();
            this.titulo=lee.next();
            this.autor=lee.next();
      }
}
public class ArchivoLibro {
      private String nomArch;
      private RegLibro rLib;
      public ArchivoLibro(String nomArch) {
            this.nomArch = nomArch;
      }
      public void crear()throws ClassNotFoundException, IOException{
            ObjectOutputStream archLib = new ObjectOutputStream(new
FileOutputStream(nomArch));
            archLib.close();
      }
      public void adicionar()throws ClassNotFoundException, IOException{
            String op;
            Scanner lee = new Scanner(System.in);
            ObjectOutputStream archLib = null;
            try{
                  if(new File(nomArch).exists())
                        archLib = new AddObjectOutputStream(new
FileOutputStream(nomArch,true));
                  else
                        archLib = new ObjectOutputStream(new
FileOutputStream(nomArch,true));
                  do{
                        rLib = new RegLibro();
                        rLib.leer();
                        archLib.writeObject(rLib);
                        System.out.print("Desea Cont. s/n");
                        op = lee.next();
                  }while(op.equals("s"));
                  archLib.close();
            }catch(Exception e){
                  System.out.print("\n FIN ADICIONA");
            }
      }
      public void listar() throws ClassNotFoundException, IOException{
            ObjectInputStream archLib =null;
            try {
                  archLib=new ObjectInputStream(new FileInputStream(nomArch));
                  while(true){
                        rLib=new RegLibro();
                        rLib = (RegLibro)archLib.readObject();
                        rLib.mostrar();
```

```
			}
		} catch (Exception e) {
			System.out.println("Fin Listado");
		}finally{
			archLib.close();
		}
	}
	public boolean Eliminar(String codMed)throws ClassNotFoundException,IOException{
		boolean sw = false;
		ObjectInputStream archLib = null;
		ObjectOutputStream aCopia = null;
		try {
			archLib = new ObjectInputStream(new FileInputStream(nomArch));
			aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
			while(true){
				rLib = new RegLibro();
				rLib = (RegLibro)archLib.readObject();
				if(rLib.getIdLibro().equals(codMed))
					sw = true;
				else
					aCopia.writeObject(rLib);
			}
		} catch (Exception e) {
			System.out.println("Fin Elimina");
		}finally{
			archLib.close();
			aCopia.close();
			File f1 = new File(nomArch);
			File f2 = new File("copia.dat");
			f1.delete();
			f2.renameTo(f1);
		}
		return sw;
	}

	public void Modificar()throws ClassNotFoundException,IOException{
		Scanner lee = new Scanner(System.in);
		System.out.println("Intr. cod a modificar");
		String op, codMed = lee.next();
		ObjectInputStream archLib = null;
		ObjectOutputStream aCopia = null;
		try {
			archLib = new ObjectInputStream(new FileInputStream(nomArch));
			aCopia = new ObjectOutputStream(new FileOutputStream("copia.dat",true));
			while(true){
				rLib = new RegLibro();
				rLib = (RegLibro)archLib.readObject();
				if(rLib.getIdLibro().equals(codMed)){
					rLib.mostrar();   System.out.println("Desea modificar s/n");
					op = lee.next();
					if(op.equals("s")){
						rLib.leer();
						rLib.mostrar();
					}
				}
				aCopia.writeObject(rLib);
			}
		} catch (Exception e) {
			System.out.println("Fin modifica");
		}finally{
			archLib.close();   aCopia.close();
			File f1 = new File(nomArch);		File f2 = new File("copia.dat");
			f1.delete();   f2.renameTo(f1);
```

```
                }
        }
}
```