

Problema 1.-

/*

*** 1.- Un Hospital necesita mostrar listas de médicos, enfermeras, personal administrativo, personal de limpieza y personal de cocina. Para esto se puede emplear:**

Se debe crear una lista para cada tipo de empleado que existe en el hospital.

a) Crear el método *adicionar* para adicionar un objeto a la lista

b) Crear el método *eliminar* para eliminar el objeto en la posición i de la lista

c) Crear el método *eliminar* para eliminar el último objeto de la lista.

** To change this template use Tools | Options | Coding | Edit Standard Headers.*

*/

using System;

namespace Hospital

{

public class Lista<R, T>

{

private R x;

private T y;

public Lista(R x,T y)

{

this.x=x;

this.y=y;

}

public void Mostrar(){

Console.WriteLine("(" +x+" "+y+"");

}

public R X {

get {

return x;

}

set {

x = value;

}

}

public T Y {

get {

return y;

}

set {

y = value;

}

}

}

}

```

/*
 * Created by SharpDevelop.
 * User: ALEXPART
 * Date: 30/11/2017
 * Time: 11:18
 *
 * To change this template use Tools | Options | Coding | Edit Standard Headers.
 */

```

```

using System;

```

```

namespace Hospital

```

```

{
    /// <summary>
    /// Description of ConjuntoLista.
    /// </summary>
    public class ConjuntoLista<W,S>
    {
        private int n;
        private Lista<W,S>[] p=new Lista<W, S>[50];
        public ConjuntoLista()
        {
            n=0;
        }
        public void AdiPunto(Punto<W,S> p){
            this.p[n]=p;
            n++;
        }

        public void OrdenarX()
        {
            Punto <W,S> Puntoaux;
            for (int i = 0; i < n-1; i++) {
                for (int j = i+1; j < n; j++) {
                    if(p[i].X.CompareTo(p[j].X)>0){
                        Puntoaux=p[i];
                        p[i]=p[j];
                        p[j]=Puntoaux;
                    }
                }
            }
        }
        public void OrdenarY()
        {
            Punto <W,S> Puntoaux;
            for (int i = 0; i < n-1; i++) {
                for (int j = i+1; j < n; j++) {

```

```

        if(p[i].Y.CompareTo(p[j].Y)>0){
            Puntoaux=p[i];
            p[i]=p[j];
            p[j]=Puntoaux;
        }
    }
}

public void Imprime(){
    for (int i = 0; i < n; i++) {
        this.p[i].Mostrar();
    }
}

public bool BuscarPunto(Punto<W,S>p){
    bool a=false;
    for (int i = 0; i < n; i++) {
        if(this.p[i].X.Equals(p.X)&&this.p[i].Y.Equals(p.Y))a=true;
    }
    return a;
}

}

public void eliminar(int i)
{
    for (int j = 0; j < n; j++) {
        if(p[j].X.Equals(i)){
            for (int k = j; k < n-1; k++) {
                p[k]=p[j+1];
            }
        }
    }
    n--;
}

public void eliminar(int i, string a)
{
    for (int j = 0; j < n; j++) {
        if(p[j].X.Equals(i)&&p[j].Y.Equals(a)){
            for (int k = j; k < n-1; k++) {
                p[k]=p[j+1];
            }
        }
    }
    n--;
}

using System;
namespace Hospital
{

```

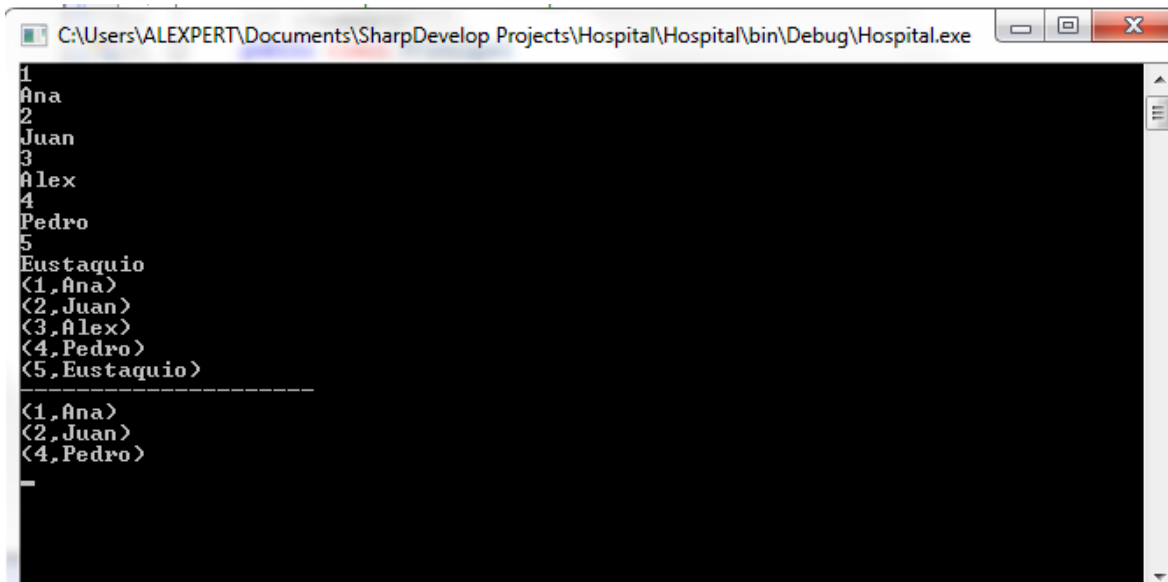
```

public class Principal
{
    public static void Main()
    {
        ConjuntoLista <int, string>c2=new ConjuntoLista<int,string>();
        for (int i = 0; i < 5; i++) { //para adicionar por teclado

            int x=Console.ReadLine()[0];
            string y=Console.ReadLine();
            Lista<int, string>p=new Lista<int, string>(x,y);
            c2.AdiPunto(p);
        }
        c2.eliminar(3);
        c2.eliminar(4,"Pedro");
    }
}

```

Consola:



```

C:\Users\ALEXPERT\Documents\SharpDevelop Projects\Hospital\Hospital\bin\Debug\Hospital.exe
1
Ana
2
Juan
3
Alex
4
Pedro
5
Eustaquio
<1, Ana>
<2, Juan>
<3, Alex>
<4, Pedro>
<5, Eustaquio>
-----
<1, Ana>
<2, Juan>
<4, Pedro>

```