1) (6 puntos) Modificar la clase ArregloEnteros para que se pueda listar, actualizar y borrar todos los datos repetidos.   
Por ejemplo: si se quiere actualizar el número 2 y hay 4 números 2 en el arreglo deberán de actualizarse los 4 con el mismo valor.  
  
a) Modificar el método BuscaDato(..)  
b) Modificar el método Actualizar(...)  
c) Modificar el método Borar(...)  
d) Modificar el método Listar(...)

public int BuscaInfo(int dato,int pos){

int i;

for(i=pos;i<=indice;i++){

if(arreglo[i]==dato){

return i;

}

}

return -1;

}

public int BuscaInfo1(int dato,int pos){

int i;

for(i=(pos+1);i<=indice;i++){

if(arreglo[i]==dato){

return i;

}

}

return -1;

}

**case 2://Borrar**

int nborrar=Teclado.entero("dame el numero a borrar \n ");

int posborrar=0;

posborrar=miarreglo.BuscaInfo(nborrar,posborrar);

if (posborrar!=-1){

do{

miarreglo.Borrar(posborrar);

posborrar=miarreglo.BuscaInfo(nborrar,posborrar);

}while(posborrar!=-1);

}

else{

System.out.println("no encuentra el dato\n");

}

break;

**case 3://Actualizar**

int buscar=Teclado.entero("El dato que desea actualizar\n");

int posbuscar=0;

int datoact=Teclado.entero("dame un nuevo valor \n");

posbuscar=miarreglo.BuscaInfo(buscar,posbuscar);

if (posbuscar==-1) {

System.out.println("No se encuentra el dato");

}

else{

while(posbuscar!=-1){

miarreglo.Actualizar(posbuscar,datoact);

posbuscar =miarreglo.BuscaInfo(buscar,posbuscar);

}

}

break;

**case 5://Buscar**

int buscar2=Teclado.entero("El dato que desea buscar\n\n");

int pos1=0;

pos1=miarreglo.BuscaInfo(buscar2,pos1);

if (pos1==-1) {

System.out.println("No se encuentra el dato");

}

else{

do{

miarreglo.Listar(pos1);

pos1=miarreglo.BuscaInfo1(buscar2,pos1);

}while(pos1!=-1);

}

break;

2) (4 puntos) También deberá de implementar el método Ordenar de la clase ArregloEnteros ejecutando el método de ordenamiento "Inserción". Para ello deberá de crear un objeto de la clase ordenamientos, o bien deberá de ejecutarlos como método estático. No podrá copiar el método en la clase ArregloEnteros, pues de lo contrario se anulará este punto. Deberá implementar el caso en el main para realizar esta acción.

public void Ordenar(){

Ordenamiento ordenar1=new Ordenamiento();

//ordenar1.Intercambio(arreglo,indice);

//ordenar1.Seleccion(arreglo,indice);

ordenar1.Insercion(arreglo,indice);

}

3)Prueba de escritorio método inserción.

public void Insercion(int []A,int indice){

int i,j,aux;

for (i=1;i<=indice;i++) {

j=i;

aux=A[i];

while(j>0&&aux<A[j-1]){

A[j]=A[j-1];

j--;

}

A[j]=aux;

}

}

}

**Inicio**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **8** | **2** | **4** | **0** | **-1** |

0 1 2 3 4

V=verdadero

F=Falso

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i** | **j** | **aux** | **j>0** | **A[j-1]>aux** | **Salida** |
| 1 | 1 | 2 | V | A[0]>2  8>2  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 8 | 8 | 4 | 0 | -1 | |
| 1 | 0 | 2 | F |  | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 2 | 8 | 4 | 0 | -1 | |
| 2 | 2 | 4 | V | A[1]>4  8>4  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 2 | 8 | 8 | 0 | -1 | |
| 2 | 1 | 4 | V | A[0]>4  2>4  F | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 2 | 4 | 8 | 0 | -1 | |
| 3 | 3 | 0 | V | A[2]>0  8>0  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 2 | 4 | 8 | 8 | -1 | |
| 3 | 2 | 0 | V | A[1]>0  4>0  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 2 | 4 | 4 | 8 | -1 | |
| 3 | 1 | 0 | V | A[0]>0  2>0  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 2 | 2 | 4 | 8 | -1 | |
| 3 | 0 | 0 | F |  | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 | 2 | 4 | 8 | -1 | |
| 4 | 4 | -1 | V | A[3]>-1  8>-1  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 | 2 | 4 | 8 | 8 | |
| 4 | 3 | -1 | V | A[2]>-1  4>-1  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 | 2 | 4 | 4 | 8 | |
| 4 | 2 | -1 | V | A[1]>-1  2>-1  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 | 2 | 2 | 4 | 8 | |
| 4 | 1 | -1 | V | A[0]>-1  0>-1  V | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 | 0 | 2 | 4 | 8 | |
| 4 | 0 | -1 | F |  | |  |  |  |  |  | | --- | --- | --- | --- | --- | | -1 | 0 | 2 | 4 | 8 | |