PROGRAMACIÓN 2º EVALUACIÓN (9/3/2017)

NOMBRE...................................................................................................................................

1.- Hacer un programa que en el **main** cree un array de cadenas que se introducirán por teclado, la dimensión también se leerá desde teclado. Desde el main se llamará a los siguientes métodos:

1. **visualizar**, nos visualizará por pantalla las cadenas que se encuentren en posiciones (0,2,4…..) en mayúsculas y las que se encuentren en las posiciones(1,3,5….) en minúsculas. (métodos: toUpperCase() ; toLowerCase() )
2. **buscar**, leerá una cadena a buscar desde teclado y determinará visualizando un mensaje, si dicha cadena se encuentra o no en el array de cadenas.
3. **ordenar,** ordenará el array de cadenas de forma ascendente

2.- El siguiente código se corresponde con una clase Almacén que permite guardar, eliminar y

listar los objetos almacenados:

**public class Almacen** {

private Object [ ] contenedor;

// El tamaño por defecto será por ejemplo 50

private int tamaño = 50;

public Almacen( ) {

// A rellenar por el alumno

}

public Almacen(int nuevoTamaño) {

// A rellenar por el alumno

}

public int numeroObjetosAlmacenados( ) {

// A rellenar por el alumno

}

public int numeroHuecos( ) {

// A rellenar por el alumno

}

public void guardar(Object objeto) {

// A rellenar por el alumno

}

public Object sacar(int indiceArray) {

// A rellenar por el alumno

}

public void listar( ){

// A rellenar por el alumno

}

}

Se pide:

1. Escriba los métodos constructores del almacén para crearlo con el tamaño por defecto y con un nuevo tamaño.

2. Implemente el método numeroObjetosAlmacenados que devolverá el número de objetos del almacén.

3. Implemente el método numeroHuecos que devolverá el número de huecos del almacén.

4. Implemente el método guardar de forma que guarde un objeto en el siguiente lugar libre. NOTA: Si no hay hueco no se guarda.

5. Implemente el método sacar para extraer un objeto en función de su posición.

NOTA: Si se saca un objeto queda un hueco.

6. Implemente el método listar que imprimirá un listado de los objetos almacenados junto con el índice asignado.

NOTA: Utilizar el método toString( ) para imprimir la información de cada objeto

3 .- Indique qué imprime el siguiente código:

class Ejercicio3 {

public static void modificarArray(int a[]){

int [] b = {2, 5, 6, 9};

a[0] = b[0];

b[1] = a[1];

a[2] = b[2];

a = b;

for (int i = 2; i < a.length; i++) System.out.println(a[i]);

}

public static void main(String [] args){

int a[] = {1, 7, 8};

modificarArray(a);

for (int i = 0; i < a.length; i++) System.out.println(a[i]);

}

}

4.- Indicar cuál sería la salida del método main de la clase Main (las clases Abuelo, Padre e Hijo se muestran a continuación)

**public class Main {**

public static void main(String[] arg){

Hijo h = new Hijo();

System.out.println("\n m1 retorna:"+ h.m1());

System.out.println("\n m2:");

h.m2();

System.out.println("\n m3:");

h.m3();

System.out.println("\n m4:" );

h.m4();

}

**}**

**public class Abuelo{**

public int m1(){

return 0;

}

public void m2(){

System.out.println(“Abuelo.m2”);

}

public void m3(){

System.out.println(“Abuelo.m3”);

}

public void m4(){

System.out.println(“Abuelo.m4”);

}

**}**

**public class Padre extends Abuelo{**

public int m1(){

return 1;

}

public void m2(){

super.m2();

System.out.println(“Padre.m2”);

}

public void m4(int i){

super.m4();

System.out.println(“Padre.m4”);

}

**}**

**public class Hijo extends Padre{**

public int m1(){

return super.m1();

}

public void m2(){

super.m2();

System.out.println(“Hijo.m2”);

}

public void m3(){

super.m3();

System.out.println(“Hijo.m3”);

}

**}**

5.-Diseñar las siguientes clases

* Una clase **Producto** con los siguientes atributos privados: String **nombre**, int **código**, double **precio,** String **marca.** La clase además tiene un constructor, un método toString() que devuelve la información del objeto y aquellos que necesites para su proceso.
* Una clase **Aplicación** que:

1. Cree una estructura de datos estática para almacenar los distintos productos que se han vendido (Se leerá desde teclado el número máximo de productos).
2. Desde un **método llenar** introduzca distintos productos en la lista hasta que le digamos que ya no queremos introducir más.
3. Muestre la lista por pantalla
4. Muestre el artículo de precio mayor.

**NOTA:** Se pide la utilización de funciones, para realizar cada apartado.