**FASE ELIMINATORIA NIVEL: JUNIOR**

1. Por ordenar se entiende el proceso de reorganizar un conjunto de objetos en una cierta secuencia de acuerdo a un criterio especificado. En general, el objetivo de este proceso es facilitar la posterior búsqueda de elementos en el conjunto ordenado. Existen múltiples ejemplos reales de conjuntos que requieren ser ordenados: la guía telefónica, índices de libros, ficheros de bibliotecas, diccionarios, ficheros de diverso tipo en oficinas, actas de exámenes, etc.

.

**DEBE REALIZAR UN ALGORITMO MEDIANTE JAVA, QUE PERMITA INGRESAR 5 DATOS EN UN VECTOR Y QUE LOS ORDENE DE MENOR A MAYOR EN CASO QUE SEAN NÚMEROS Y DE LA (Z) A LA (A) EN CASO QUE SEAN LETRAS. DEBE MOSTRAR EN PANTALLA EL RESULTADO.**

**Nota: Debe tener en cuenta si son iguales.**

**Ejemplo de entrada**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **4** | **3** | **5** | **10** | **1** | **3** |

**Ejemplo de salida**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1** | **3** | **3** | **4** | **5** | **10** |

1. **Realizar un algoritmo que simule el comportamiento de un reloj digital. Se debe imprimir la hora., los minutos y los segundos mediante consola en formato militar.**

**Ejemplo de salida.**

13:05:18

13:05:19

13:05:20

13:05:21

13:05:22

13:05:23

13:05:24

13:05:25

1. **Ingresar una cadena de caracteres cuya suma total sea divisible entre 4. Usted debe dividir la cadena y guardar en un vector, matriz o lista, (solo un tipo de arreglo) un conjunto de cuatro letras por cada una de las posiciones del arreglo seleccionado. Al finalizar debe imprimir la cadena de caracteres original y el vector con los conjuntos de letras divididos previamente.**

**Ejemplo de entrada**

|  |
| --- |
| **XCVBNMKLJHGF** |

**Ejemplo de salida**

**Cadena Original: XCVBNMKLJHGF**

|  |  |  |
| --- | --- | --- |
| **XCVB** | **NMKL** | **JHGF** |

**Divisiones**

**Rúbrica de evaluación FASE ELIMINATORIA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Información del grupo** | | | | |
| 1. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** 2. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** 3. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | **Nombre del grupo: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Tiempo Utilizado (minutos): \_\_\_\_\_**  **Bono (puntos): \_\_\_\_\_** | | |
| **EJERCICIO** | **Patronus (3)** | | **Dementor (0)** | **Puntos** |
| **#1 Ordenar vector.** | **Permite ingresar 5 datos en un vector, los ordena de menor a mayor en caso que sean números y de la (z) a la (a) en caso que sean letras y muestra el resultado en pantalla** | | **No compila** |  |
| **#2 Reloj Digital.** | **Muestra por consola el comportamiento de un reloj digital. Se evidencia la impresión de horas minutos y los segundos en formato militar.** | | **No compila** |  |
| **#3 Split** | **Permite ingresar una cadena de caracteres cuya suma total sea divisible entre 4. El programa divide la cadena y la guarda en un vector, matriz o lista, (solo un tipo de arreglo), imprimiendo la cadena de caracteres original y el vector con los conjuntos de letras divididos previamente.** | | **No compila** |  |
|  | | | **PUNTOS TOTALES** |  |

**PRUEBA FASE FINAL**

**Realizar un CRUD (Consultar, Registrar, modificar y eliminar) con interfaz gráfica, que permita gestionar la información de un cliente, en la cual se solicitan los siguientes datos: Nombre, identificación, correo electrónico y teléfono. Para este caso debe almacenar la información en vectores, en el caso de eliminar debe remplazar el valor por un campo vacío o un cero en caso que sea numérico.**

**Nota: La programación es libre, puede ser estructurada u orientada a objetos. No se tendrá en cuenta el diseño.**

**Rúbrica de evaluación FASE FINAL**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Información del grupo** | | | | |
| 1. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** 2. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** 3. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | **Nombre del grupo: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Tiempo Utilizado (minutos): \_\_\_\_\_**  **Bono (puntos): \_\_\_\_\_** | | |
| **CRITERIO** | **Descripción** | | **Ponderación** | **Puntos** |
| **#1 CRUD COMPLETO** | **El programa consulta, registra, modifica y elimina la información completa del cliente** | | **3** |  |
| **#2 CRUD INCOMPLETO** | **El programa realiza como mínimo dos funciones del CRUD** | | **1** |  |
|  | | | **PUNTOS TOTALES** |  |