

Algoritmos y Estructura de Datos – TI

Guía de Problemas - Semana 06

Arreglo lineal: ingreso personalizado, ampliación y modificación.

Problema_6_1

Diseñe la clase **ArregloEdades** en el paquete **semana_06** con el atributo privado **edad** (*int*) de tipo arreglo lineal y el atributo privado **indice** (*int*).

Implemente además:

- Un Constructor sin parámetros que reserve 10 espacios en **edad** e inicialice con 0 al **indice**.
- Un método **tamano** que retorne la cantidad de datos ingresados hasta ese momento.
- Un método **obtener** que reciba una posición y retorne la edad registrada en dicha posición.
- Un método privado **ampliarArreglo** que extienda el arreglo en diez espacios más.
- Un método **adicionar** que reciba una edad y la registre en la posición que corresponda. Verifique primero si el arreglo está lleno para invocar al método **ampliarArreglo**.
- Un método **eliminarAlFinal** que elimine lógicamente la última edad del arreglo.
- Un método **eliminarTodo** que elimine lógicamente todas las edades.
- Un método **edadMayor** que retorne la mayor de todas las edades.
- Un método **posPrimeraEdadAdulta** que retorne la posición de la primera edad encontrada en el rango de 20 a 59 años. En caso no exista retorne -1.
- Un método **incrementarPrimeraEdadAdulta** que aumente la primera edad adulta en 5 años.
- Un método **reemplazarPrimeraEdadAdulta** que cambie la primera edad adulta por la edad mayor.

En la clase principal:

- Declare y cree el objeto global **ae** de tipo **ArregloEdades**.
- Implemente un método **listar** que visualice las edades registradas hasta ese momento.
- A la pulsación del botón **Adicionar** lea una edad por GUI y adiciónela al arreglo. Invoque luego al método **listar**.
- A la pulsación del botón **Eliminar al final** invoque al método **eliminarAlFinal** e invoque al método **listar**. En caso que el arreglo esté vacío muestre el mensaje respectivo.
- A la pulsación del botón **Eliminar todo** invoque al método **eliminarTodo**. En caso que el arreglo esté vacío muestre el mensaje respectivo.
- A la pulsación del botón **Incrementar primera edad adulta** invoque al método **incrementarPrimeraEdadAdulta**. En caso que no exista ninguna edad adulta visualice un mensaje al respecto.
- A la pulsación del botón **Reemplazar primera edad adulta** invoque al método **reemplazarPrimeraEdadAdulta**. En caso que no exista ninguna edad adulta visualice un mensaje al respecto.

Problema_6_2

Diseñe la clase **ArregloNotas** en el paquete **semana_06** con el atributo privado **nota** (*int*) de tipo arreglo lineal y el atributo privado **indice** (*int*).

Implemente además:

- Un Constructor sin parámetros que reserve 10 espacios en **nota** e inicialice con 0 al **indice**.
- Un método **tamaño** que retorne la cantidad de datos ingresados hasta ese momento.
- Un método **obtener** que reciba una posición y retorne la nota registrada en dicha posición.
- Un método privado **ampliarArreglo** que extienda el arreglo en diez espacios más.
- Un método **adicionar** que reciba una nota y la registre en la posición que corresponda. Verifique primero si el arreglo está lleno para invocar al método **ampliarArreglo**.
- Un método **eliminarAlFinal** que elimine lógicamente la última nota del arreglo.
- Un método **eliminarTodo** que elimine lógicamente todas las notas.
- Un método **posUltimaNotaDesaprobatoria** que retorne la posición de la última nota menor que 13. En caso no exista retorne -1.
- Un método **decrementarUltimaNotaDesaprobatoria** que disminuya la última nota menor que 13 en dos puntos. En caso que al disminuir resulte una nota negativa, fije la nota en cero.
- Un método **reemplazarUltimaNotaDesaprobatoria** que cambie la última nota desaprobatoria por la última nota del arreglo.

En la clase principal:

- Declare y cree el objeto global **an** de tipo ArregloNotas.
- Implemente un método **listar** que visualice las notas registradas hasta ese momento.
- A la pulsación del botón **Adicionar** lea una nota por GUI y adiciónela al arreglo. Invoque luego al método listar.
- A la pulsación del botón **Eliminar al final** invoque al método **eliminarAlFinal** e invoque al método **listar**. En caso que el arreglo esté vacío muestre el mensaje respectivo.
- A la pulsación del botón **Eliminar todo** invoque al método **eliminarTodo**. En caso que el arreglo esté vacío muestre el mensaje respectivo.
- A la pulsación del botón **Decrementar última nota desaprobatoria** invoque al método **decrementarUltimaNotaDesaprobatoria**. En caso que no exista ninguna nota desaprobatoria visualice un mensaje al respecto.
- A la pulsación del botón **Reemplazar última nota desaprobatoria** invoque al método **reemplazarUltimaNotaDesaprobatoria**. En caso que no exista nota desaprobatoria visualice un mensaje al respecto.

Propuesto_6_1

Diseñe la clase **ArregloTemperaturas** en el paquete **semana_06** con el atributo privado **temperatura** (*double*) de tipo arreglo lineal y el atributo privado **indice** (*int*).

Implemente además:

- Un Constructor sin parámetros que reserve 10 espacios en **temperatura** e inicialice con 0 al **indice**.
- Un método **tamano** que retorne la cantidad de datos ingresados hasta ese momento.
- Un método **obtener** que reciba una posición y retorne la temperatura registrada en dicha posición.
- Un método privado **ampliarArreglo** que extienda el arreglo en diez espacios más.
- Un método **adicionar** que reciba una temperatura y la registre en la posición que corresponda. Verifique primero si el arreglo está lleno para invocar al método **ampliarArreglo**.
- Un método **eliminarAlFinal** que elimine lógicamente la última temperatura del arreglo.
- Un método **eliminarTodo** que elimine lógicamente todas las temperaturas.
- Un método **temperaturaMenor** que retorne la temperatura más baja.
- Un método **posPrimeraTemperaturaNormal** que retorne la posición de la primera temperatura encontrada en el rango de 36.1°C a 37.2°C.
- Un método **reemplazarPrimeraTemperaturaNormal** que cambie la primera temperatura normal por la menor temperatura del arreglo. Para el efecto, invoque a los métodos **posPrimeraTemperaturaNormal** y **temperaturaMenor**.
- Un método **incrementarTemperaturas** que aumente todas las temperaturas en 0.2°C.

En la clase principal:

- Declare y cree el objeto global **at** de tipo **ArregloTemperaturas**.
- Implemente un método **listar** que visualice las temperaturas registradas hasta ese momento.
- A la pulsación del botón **Adicionar** lea una temperatura por GUI y adiciónela al arreglo. Invoque luego al método **listar**.
- A la pulsación del botón **Eliminar al final** invoque al método **eliminarAlFinal** e invoque al método **listar**. En caso que el arreglo esté vacío muestre el mensaje respectivo.
- A la pulsación del botón **Eliminar todo** invoque al método **eliminarTodo**. En caso que el arreglo esté vacío muestre el mensaje respectivo.
- A la pulsación del botón **Reemplazar primera temperatura normal** invoque al método **reemplazarPrimeraTemperaturaNormal**. Visualice los cambios. En caso que no exista ninguna temperatura normal visualice un mensaje al respecto.
- A la pulsación del botón **Incrementar temperaturas** invoque al método **incrementarTemperaturas**. Visualice los cambios.

"Ser tenaz en nuestro trabajo es hacerlo con esfuerzo, ahínco y dedicación."

MP