

| | | | | | |
|--------------------|--|--------------|-------|--------------|------------|
| EVALUACION | Parcial AED1 | GRUPO | TODOS | FECHA | 12/02/2025 |
| MATERIA | Algoritmos 1 | | | | |
| CARRERA | Analista Programador / Analista en Tecnologías de la Información | | | | |
| CONDICIONES | - Puntos: 45 - Duración: 2 horas - Sin material | | | | |

Ejercicio 1 (15 pts)

Dado una matriz cuadrada de enteros, no repetidos:

| | | | |
|----|---|----|----|
| 1 | 6 | 7 | 31 |
| 10 | 4 | 11 | 40 |
| 3 | 4 | 5 | 43 |
| 51 | 2 | 8 | 9 |

- a) Escriba un algoritmo que retorne el máximo de los elementos de la matriz, entre dos filas dadas. **(5 pts)**

//pre: 0 <= fila1 && fila1 < mat.length && 0 <= fila2 && fila2 < mat.length

Firma: **public static int minimoEntreCol (int[] [] mat, int fila1, int fila2)**

Ej: para fila1: 1 y fila2:2, el resultado debería ser: 43

- b) Escriba un método recursivo que muestre los elementos de la diagonal principal de la matriz, en forma inversa. Realice el diagrama de llamadas para la matriz dada. **(10 pts)**

Ej: para el ejemplo de la matriz dada, se debería mostrar 9 5 4 1

Ejercicio 2 (10 pts)

Dado un vector ordenado en forma ascendente de números enteros y un valor, implemente el método de búsqueda por punto medio (en forma iterativa o recursiva)

Firma: **public static boolean buscar (int[] vec, int valor)**

Ejercicio 3 (20 pts)

Se ha implementado una Lista simplemente enlazada que cuenta con un puntero al inicio, un entero para almacenar la cantidad de elementos y un entero que limita su capacidad máxima.

```
public class Lista {          public class Nodo {
    private Nodo inicio;      private int dato;
    private int cantidad;     private Nodo sig;
    private int capMax;
    //.....                //Métodos de acceso y modificación disponibles
}                             }
```

Implemente las siguientes operaciones en el TAD Lista:

- a) Implemente la operación de instancia **agregarInicio**, que inserta el elemento al comienzo de la lista, retornando un boolean que indica si se pudo efectivamente agregar **(10 pts)**

Firma: **public boolean agregarInicio(int dato)**

- b) Implemente la operación de instancia **colaMayoresDe** (de forma recursiva), que retorne una cola con todos los elementos mayores al parametro. **(10 pts)**

Firma: **public Cola colaMayoresDe (int dato)**

Ej: para la lista 10-3-5-76-11-4-2 y el dato 4, debería retornar la siguiente cola: 10 5 76 11 (frente)

Nota: se pueden crear métodos auxiliares si lo considera necesario pero se debe implementar

Notas:

- Para todas las operaciones solicitadas se dispone de gets y sets
- Se cuentan con las operaciones de Cola y Pila vistas en el curso.
- Se pueden utilizar funciones o métodos auxiliares, pero se deben implementar.
- Indicar claramente que parte se está resolviendo.
- Escribir con letra legible ya que se considerará durante la corrección.