

EVALUACION	Parcial Matutino	GRUPO	TODOS	FECHA	XX/07/2024
MATERIA	Algoritmos 1				
CARRERA	Analista Programador / Analista en Tecnologías de la Información				
CONDICIONES	- Puntos: 45 - Duración: 2 horas - Sin material				

Ejercicio 1 (10 pts)

Sea M una matriz de enteros:

Ejemplo de matriz de 4 X 5:

	0	1	2	3	4
0	1	6	3	11	3
1	11	12	6	40	14
2	5	5	6	2	6
3	8	6	20	1	9
4	7	23	1	5	4
5	6	8	2	4	6

- a) desarrollar un algoritmo que verifique que indique si existen en la matriz 2 columnas que sumen exactamente lo mismo. En caso de existir, retornar true; en caso contrario retornar false. **(5 puntos)**

Ejercicio 2 (8 pts)

A continuación, se detalla una versión del algoritmo selection sort, el cual cuenta con algunos errores en la sección marcada en amarillo.

```
public static void selectionSort(int[] arr) {
    for (int i = 0; i < arr.length; i++) {
        int arr[minIdx] = i;
        for (int j = i + 1; j < arr.length-1; j++) {
            if (arr[j] > arr[minIdx]) {
                minIdx = j;
            }
        }
        // Completar el siguiente intercambio
        int temp = arr[minIdx];
        arr[minIdx] = arr[i];
        arr[i] = temp;
    }
}
```

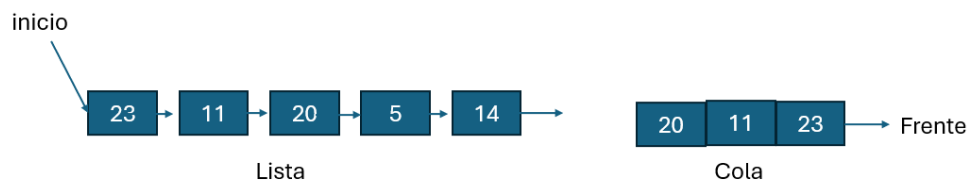
- a) Corrija únicamente lo que considere que es incorrecto del algoritmo. **(5 puntos)**
b) Para un vector con los valores [34,1,22,18,33], muestre como quedaría el vector luego de cada iteración de búsqueda del mínimo. **(3 puntos)**

Ejercicio 3 (12 pts)

Dada una lista simplemente enlazada de nodos con valores enteros positivos, una cantidad C y una cola vacía:

- a) Implemente un algoritmo recursivo que coloque los primeros C elementos de la lista en la cola, de forma de que el primer elemento de la lista quede en el frente de la cola. Defina las pre y post condiciones. **(10 pts)**. Se sumen disponibles los métodos gets y sets, así como todas las operaciones del TAD Cola (frente(), encolar(), desencolar(), esVacia())

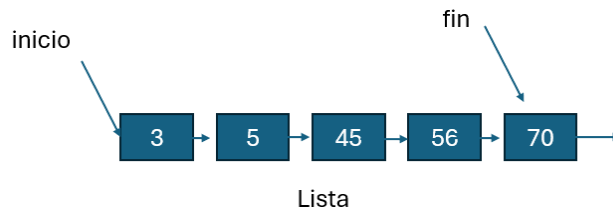
Ejemplo para un valor C = 3



- b) Realice el diagrama de llamadas **(2 pts)**

Ejercicio 4 (15 pts)

Dada una implementación de un TAD Lista con nodos simplemente enlazados ordenados en forma ascendente, con puntero al inicio y puntero al final como se muestra a continuación:



- a) Implemente una nueva operación en el TAD que, dado dos posiciones, retorne una nueva lista con los elementos que se encuentran dentro de esas dos posiciones (se asumen las posiciones correctas). Ejemplo: para una pos1:2 y una pos2:4, debería retornar una sublista con: 5->45->56

Firma sugerida: public Lista subLista(int pos1, int pos2) (8 pts)

- b) Implemente una nueva operación en el TAD que, dado un vector de enteros y una posición de la lista, inserta desde dicha posición todos los elementos del vector.

Ejemplo: para un vector [7,11] y una posición: 2, la lista quedaría con: 3->7->11->5->45->56->70

Firma sugerida: public void insertarVectorEnLista(int v[], int pos) (7 pts)

Si se utilizan operaciones del TAD no solicitadas, deben implementarse. Se cuentan disponibles los métodos gets y sets.