

EVALUACION	Examen	FECHA	20/10/2017
MATERIA	Algoritmos y Estructuras de Datos1		
CARRERA	Analista Programador / Analista en Tecnologías de la Información		
CONDICIONES	<p>Especifique las pre y post condiciones de los métodos que implemente.</p> <p>Serán tenidos en cuenta ejercicios o partes de ellos completas y que el estilo y metodología de desarrollo se ajuste al curso.</p> <p>IMPORTANTE</p> <ul style="list-style-type: none"> - Duración 3 horas - NO se puede consultar material 		

ESTUDIANTE (nombre y número)		Nota
--	--	-------------

Ejercicio 1 (35 puntos)

Dado el siguiente array **V**:

1	5	17	11	21	13	15	19	23	25
---	---	----	----	----	----	----	----	----	----

0 1 2 3 4 5 6 7 8 9

- a) Implementar un algoritmo que haga una transposición entre los k primeros términos del array y los n – k términos restantes sin utilizar un vector auxiliar.

Firma para el método a implementar :

- public void cambiar(int[] array, int n, int k)

Si ejecutamos el método con los valores del ejemplo, Cambiar(V, 3, k) luego de ejecutado el algoritmo, el vector resultante es el siguiente:

11	21	13	15	19	23	25	1	5	17
----	----	----	----	----	----	----	---	---	----

0 1 2 3 4 5 6 7 8 9

- b) Ordenar el vector utilizando el método de inserción
c) Implemente un algoritmo recursivo que sume los elementos del vector

Utilizar la siguiente firma:

int suma_vec(int v [], int n) // n es la cantidad de elementos del vector

Ejercicio 2 (25 puntos)

Dados los siguientes pseudocódigos correspondientes a los métodos de ordenamiento

<pre>void _____Sort (int* A, int n){ int i, j, First; for (i = 1; i < n; i++){ First = A[i]; j = i-1; while (j >= 0 && First < A[j]){ A[j+1] = A[j]; j--; } A[j+1] = First; } }</pre>	<pre>void _____Sort (int* A, int n){ int i, j, posmin, tmp; for (i = 0; i < n-1; i++){ posmin = i; for (j = i+1; j < n; j++){ if (A[j] < A[posmin]) posmin = j; } // Intercambio de elementos tmp = A[i]; A[i] = A[posmin]; A[posmin] = tmp; } }</pre>
--	---

- Indique a que algoritmo corresponde cada pseudocódigo.
- Aplique los mismos al siguiente vector de enteros 2, 3, 5, 1, 4. Y:
 - Muestre la secuencia generada hasta ordenar el vector.
 - Determine cual algoritmo emplea menos *esfuerzo* en ordenarlo.

Para ello se consideran las siguientes reglas:

- La comparación entre elementos del arreglo implica *una unidad de esfuerzo*
- El intercambio de elementos en el arreglo implica *dos unidades de esfuerzo* (solo si son elementos diferentes).

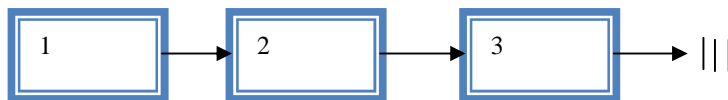
Justifique su respuesta mostrando paso a paso la ejecución de los algoritmos, especificando en cada paso qué elementos se comparan, cuáles intercambios se realizan y cómo queda el arreglo al final de cada paso.

Ejercicio 3 (40 puntos)

Dada la siguiente programa en el que se esta probando una lista simplemente enlazada:

```
public static void main(String[] args) {  
  
    System.out.println("Creamos lista vacia:");  
    Lista miLista=new Lista();  
    System.out.println("Agregamos al inicio valores 3, 2, 1:");  
    miLista.agregarInicio(3);  
    miLista.agregarInicio(2);  
    miLista.agregarInicio(1);  
    System.out.println("Mostrar Lista: ");  
    miLista.mostrar (miLista.getInicio()); // se debe mostrar en orden inverso del ultimo al primero  
}
```

Con la siguiente estructura



- A) Defina la estructura de nodo y lista para una lista de enteros.
- B) Implemente un método agregar inicio
- C) Implemente un método que muestre la lista del ultimo al primero
Es decir, si se ingresan los datos como se muestra en la figura y el código el método mostrar debe mostrar 3 – 2 - 1