DOCUMENTO TÉCNICO

# Requisitos

## Especificación

#RESUMEN DEL PROBLEMA (opcional)

Buscar el modo de ordenar los números de una lista A

### Entrada:

### Una lista A con una extensión de n números enteros en desorden

### Salida:

### Una lista A con una extensión de n números enteros organizados de menor a mayor

# Diseño

## Estrategia

Para ordenar la lista de números enteros, se toma el elemento más pequeño (el cual va a estar en la primera posición) y ya teniendo este, se va tomando número a número, repitiendo el proceso de tomar el menor y se van insertando en las posiciones siguientes de la lista. Para esto se toma el primer mínimo, se inserta en la lista y se repite el mismo proceso con los mismos números, pero quitando el numero insertado anteriormente.

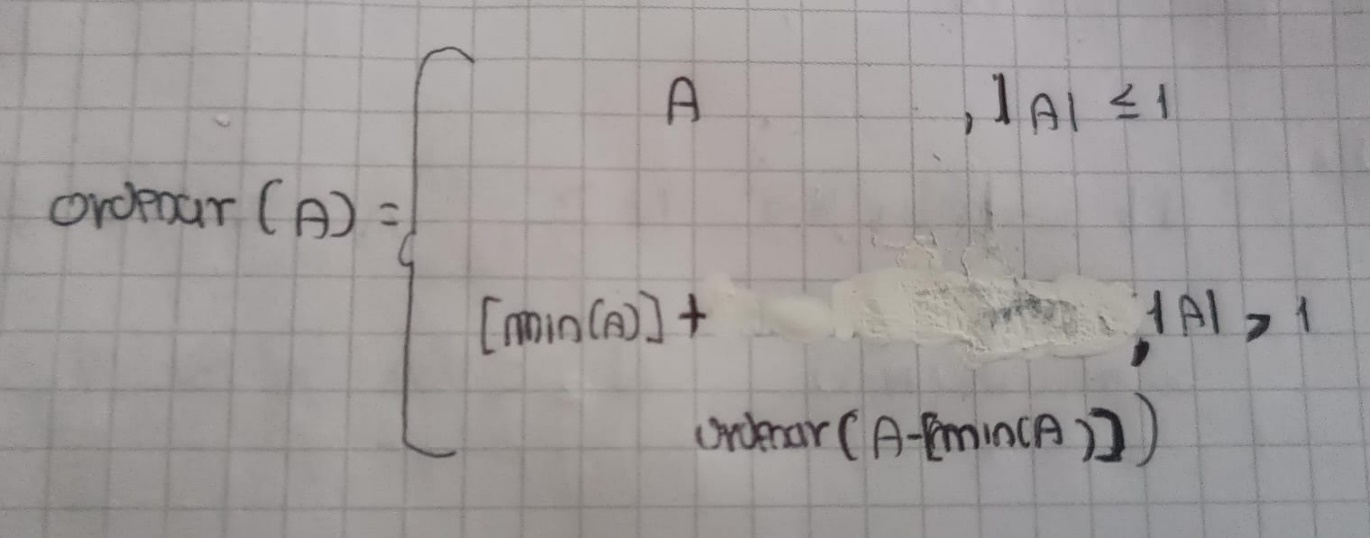
Ya teniendo la idea clara del código, se procede a crear la función a trozos la cual va a tener dos condiciones:

1. Si la longitud de la lista A sea menor o igual a 1, va a retornar el numero contenido
2. Si la longitud de la lista A sea mayor a 1, va a realizar el proceso mencionado anteriormente, de la solución del problema

#ESTRUCTURA DE DATOS

Se usaron las listas para almacenar los datos de los números enteros

#ALGORITMO

**

## Casos de prueba

|  |  |  |
| --- | --- | --- |
| **Entrada** | **Justificación** | **Salida** |
|  |  |  |
|  | Entrada vacia |  |
| 1 | Entrada con un solo numero | 1 |
| 1,4,3,2 | Caso común | 1,2,3,4 |

# Análisis

## Temporal

Con el código aplicado en Python de la manera mas optima y eficiente y por estimación, se puede observar que el costo del ordenamiento es de n.

# Código

*Al igual que el punto anterior pueden guiarse con este ejemplo*

## Documentación

Dentro del código.

## Fuentes

*Las fuentes son el código sobre el cual escribieron el documento*

/arena\_1\_a