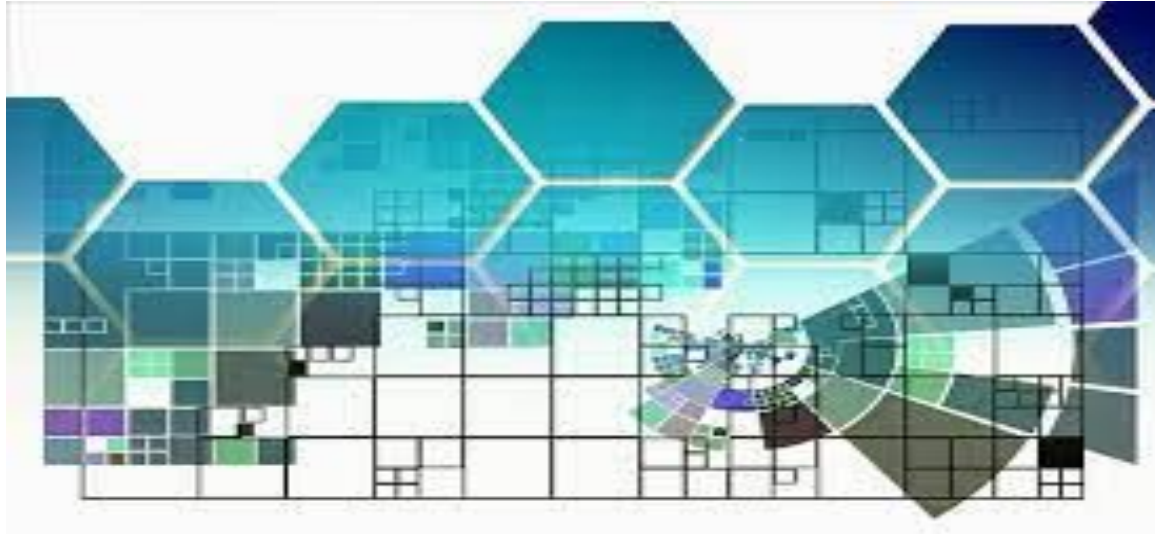


Estructura de Datos y Algoritmos II



Algoritmos de Ordenamiento.

Bubble Sort.

Ing. Mercedes Aguilar Chagoyán

Contenido.

- I. Objetivo.
- II. Repaso de Ordenamiento.
- III. Revisión de ejercicios.
 - A. Ejercicios de repaso (examen diagnóstico).
 - B. Ejercicio de Ordenamiento.
- IV. Bubble Sort.
- V. Ejercicios.
- VI. Lecturas y ejercicios del lenguaje de programación Python.
- VII. Fuentes.



Objetivo.

El alumno aprenderá, comprenderá y pondrá en práctica los conceptos adquiridos en la clase.



Repaso.

Responde las siguientes preguntas:

1. ¿Qué es ordenamiento?
2. ¿Qué es una pila?
3. ¿Qué es la recursividad?
4. Menciona algunos de los métodos de ordenamiento directos.
5. ¿Cuántos tipos de ordenamiento existen?



Revisión.

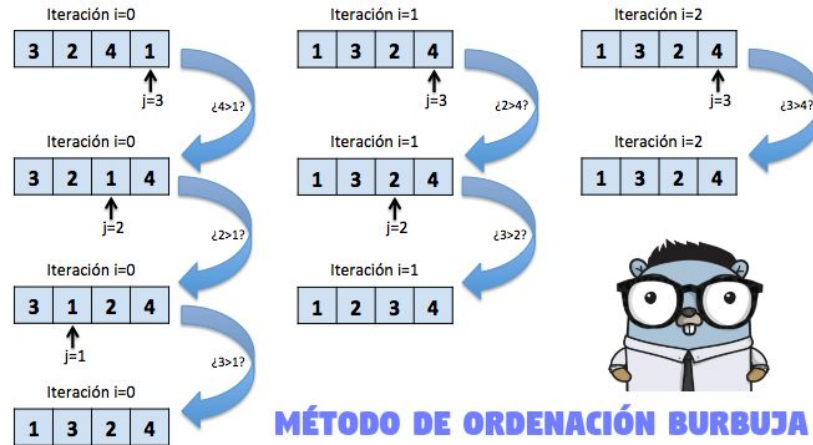
Explica la solución dada a cada uno de los ejercicios de la clase pasada.

1. Repaso.
2. Ordenamiento.



¿Qué es Bubble Sort?

Es un algoritmo que ordena los elementos contenidos en un arreglo "array".



Funcionamiento del método burbuja “*Bubble Sort*”.

Vector original

45	17	23	67	21
----	----	----	----	----

Iteración 1:

45	17	23	67	21
----	----	----	----	----

Se genera cambio

17	45	23	67	21
----	----	----	----	----

Se genera cambio

17	23	45	67	21
----	----	----	----	----

No hay cambio

17	23	45	67	21
----	----	----	----	----

Se genera cambio

17	23	45	21	67
----	----	----	----	----

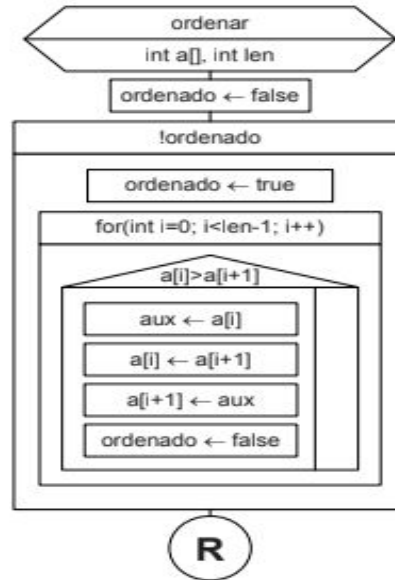
Fin primera iteración

Ejercicio 1.

1. Explica con tus propias palabras lo que entiendes por el método de la burbuja.
2. Contesta estas preguntas:
 - a. ¿Es un método eficiente?
 - b. ¿Es un algoritmo fácil de implementar?
 - c. ¿Cómo funciona este método?



Diagrama de flujo.



Ejercicio 2.

De acuerdo al diagrama de flujo, realiza el algoritmo para resolver el método.



El método burbuja.

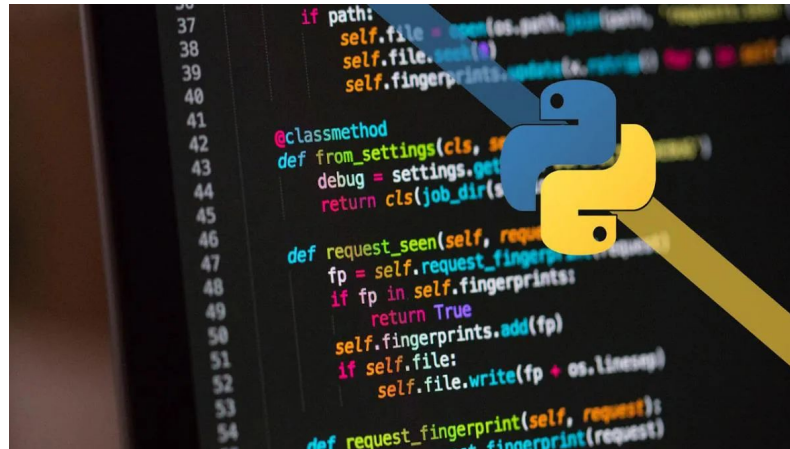
Bubble Sort o “” sigue el enfoque de clasificación de intercambio. Es muy fácil de implementar, pero tiende a ser particularmente lento para correr. Supongamos que tenemos una matriz de tamaño n que deseamos ordenar. Burbuja La clasificación comienza comparando un $[n-1]$ con un $[n-2]$ y los intercambia si están en el orden incorrecto. Luego compara un $[n-2]$ y un $[n-3]$ y los intercambia si es necesario, y así sucesivamente. Esto significa que una vez que llegue a $[0]$, la entrada más pequeña estará en el lugar correcto. Entonces comienza desde la parte de atrás de nuevo, comparando pares de "vecinos", pero dejando la entrada cero sola (que es se sabe que es correcto). Después de que haya llegado al frente nuevamente, la segunda entrada más pequeña será en su lugar. Sigue haciendo "pases" sobre la matriz hasta que se ordena. De manera más general, en el i -ésimo stage Bubble Sort compara las entradas vecinas "desde atrás", intercambiándose según sea necesario. El elemento con el índice más bajo que se compara con su vecino derecho es un $[i-1]$. Después de la i th etapa, las entradas a $[0]$, ..., a $[i-1]$ están en su posición final.

Python.

¿Qué es python?

¿Para qué es utilizado este lenguaje?

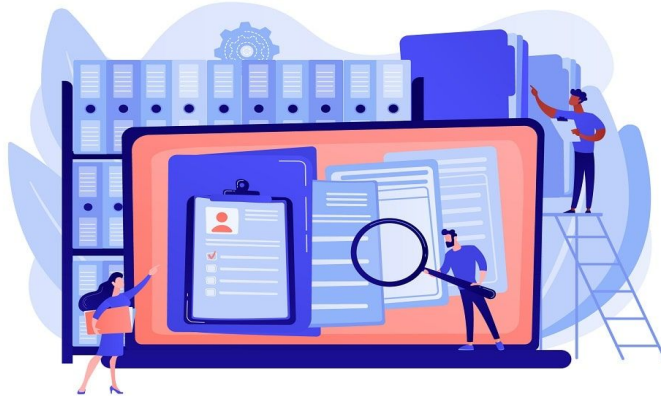
Realiza un entorno virtual de python nombrandolo “HolaMundo”.



Fuentes.

Libro 1: “Data Structures and Algorithms”

Libro 2: “Algoritmos a fondo con implementación en C y Java”.



¡Gracias por su atención!



El avance de la ciencia y la tecnología no implica creciente complejidad intelectual en la vida de la mayoría de la gente. A menudo significa lo contrario.

–Thomas Sowell