

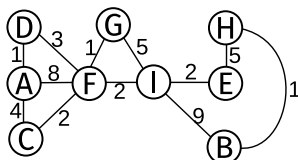
Algoritmos 2, Curso Mendez ~ 3er Final, 2do Cuatrimestre 2024 ~ 2025-02-06

Apellido y nombre: _____

Padrón: _____ Modalidad: Completo / Reducido

Nota final:				

- 1) Escriba una función (en **C99** o **Python**) como parte de la implementación de un árbol binario que invierta el árbol. Explique cómo es la estructura del árbol y cómo funciona el algoritmo utilizando gráficos. Muestre un ejemplo del funcionamiento.
- 2) Ordene el siguiente vector de menor a mayor utilizando **Mergesort**. Muestre cada paso del algoritmo. Justifique la complejidad. $V = [5, 9, 6, 1, 2, 3, 8, 7, 0]$
- 3) Explique qué es el algoritmo de **Dijkstra** y para qué sirve. Aplíquelo al siguiente grafo mostrando el resultado de cada paso y el resultado final comenzando desde **A**.



- 4) Dado el grafo del punto 3, sobreescriba los pesos de las aristas para que todas sean 1. Escriba (en **C99** o **Python**) un algoritmo que imprima por pantalla los elementos de este nuevo grafo ordenados por distancia total creciente desde **A**. Explique cómo funciona el algoritmo y muestre la salida para el grafo. ¿Es única la solución? ¿Por qué?
- 5) Explique qué es un **diccionario** y cómo difiere de una **tabla de hash**. Escriba en (**C99** o **Python**) el código para insertar un elemento en una tabla de hash abierta. Explique los puntos importantes de esta función y cómo funciona.

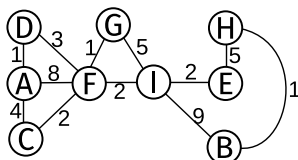
Algoritmos 2, Curso Mendez ~ 3er Final, 2do Cuatrimestre 2024 ~ 2025-02-06

Apellido y nombre: _____

Padrón: _____ Modalidad: Completo / Reducido

Nota final:				

- 1) Escriba una función (en **C99** o **Python**) como parte de la implementación de un árbol binario que invierta el árbol. Explique cómo es la estructura del árbol y cómo funciona el algoritmo utilizando gráficos. Muestre un ejemplo del funcionamiento.
- 2) Ordene el siguiente vector de menor a mayor utilizando **Mergesort**. Muestre cada paso del algoritmo. Justifique la complejidad. $V = [5, 9, 6, 1, 2, 3, 8, 7, 0]$
- 3) Explique qué es el algoritmo de **Dijkstra** y para qué sirve. Aplíquelo al siguiente grafo mostrando el resultado de cada paso y el resultado final comenzando desde **A**.



- 4) Dado el grafo del punto 3, sobreescriba los pesos de las aristas para que todas sean 1. Escriba (en **C99** o **Python**) un algoritmo que imprima por pantalla los elementos de este nuevo grafo ordenados por distancia total creciente desde **A**. Explique cómo funciona el algoritmo y muestre la salida para el grafo. ¿Es única la solución? ¿Por qué?
- 5) Explique qué es un **diccionario** y cómo difiere de una **tabla de hash**. Escriba en (**C99** o **Python**) el código para insertar un elemento en una tabla de hash abierta. Explique los puntos importantes de esta función y cómo funciona.