

## Taller Pre-Parcialito 3 [23/05] - Jueves

1. Realizar el seguimiento del algoritmo de *heapsort* para ordenar el siguiente arreglo: [28, 49, 56, 98, 70, 63, 112, 14, 21, 7]

¿De qué orden es este algoritmo?

2. Escribir una *función* en C, que dado un arreglo de N cadenas y un entero positivo K, devuelva una lista con las K cadenas más largas. Se espera que el orden del algoritmo sea  $O(n \log k)$ . Justificar el orden.

3. Hacer el seguimiento de las siguientes operaciones en un heap de mínimos inicialmente vacío, mostrando en cada paso cómo queda el heap representado como un arreglo: encolar 4, encolar 5, encolar 8, encolar 9, desencolar, encolar 2, encolar 3, desencolar, desencolar, desencolar.

**Nota:** podés ayudarte representando el estado del heap en su forma de árbol.

4. Proponga un grafo no conexo de al menos 10 vértices que cumpla con las características para poder aplicarle un orden topológico:

A) Escriba 2 ordenes topológicos distintos para el grafo propuesto. B) Exprese al grafo como matriz de incidencia, matriz de adyacencia y lista de adyacencia.

5. Un árbol no dirigido es un grafo no dirigido que cumple con las siguientes propiedades:

- $|E| = |V| - 1$
- Es Acíclico
- Es Conexo

Por teorema, si un grafo cumple dos de estas propiedades, será árbol (y por consiguiente, cumplirá la tercera). Haciendo uso de esto (y únicamente de esto), se pide implementar una función que reciba un grafo implementado con lista de adyacencias y determine si se trata o no de un árbol.

6. Dado un grafo no dirigido, determinar si se pueden utilizar sólo dos colores para pintar cada vértice, tal que ningún vértice tenga el mismo color que sus adyacentes.

Indicar el orden del algoritmo implementado. ¿Qué tipo de recorrido utilizaste? Justificar.

**Nota:** Por simplificación, podés suponer que el grafo es conexo.

7. Implementar una *función* que dado un grafo dirigido G (implementado con listas de adyacencias), permita obtener el grado de entrada de todos los vértices (por ejemplo, devolviendo un diccionario), en orden  $O(V+E)$ .

8 Matías está en Barcelona y quiere recorrer un museo. Su idea es hacer un recorrido bastante lógico: empezar en una sala (al azar), luego ir a una adyacente

a ésta, luego a una adyacente a la segunda (si no fue visitada aún), y así hasta recorrer todas las salas. Cuando no tiene más salas adyacentes para visitar (porque ya fueron todas visitadas), simplemente vuelve por donde vino buscando otras salas adyacentes.

Teniendo un grafo no dirigido, que representa el mapa del museo (donde los vértices son salas, y las aristas  $(v, w)$  indican que las salas  $v$  y  $w$  se encuentran conectadas), implementar un algoritmo que nos devuelva una lista con un recorrido posible de la idea de Matías para visitar las salas del museo.

Indicar el recorrido utilizado y el orden del algoritmo. Justificar.