

Algoritmos y Programación II (75.41, 95.15) – Curso Buchwald

3.er parcialito – 02/07/2021

Resolvé los siguientes problemas en forma clara y legible. Podés incluir tantas funciones auxiliares como creas necesarias.

Debés resolver 3 de los siguientes ejercicios. Los ejercicios a resolver están definidos por tu padrón. Podés revisar en [esta planilla](#).

1. Dado un juego de 2 jugadores, donde cada jugador tiene N cartas y gana quien posea las K cartas cuya suma sea mayor. Se desea implementar una función:

```
int obtener_ganador(int* cartas_1, int* cartas_2, size_t n, size_t k);
```

Donde `cartas_1` representa un array con los valores de las cartas del jugador 1, `cartas_2` representa un array con los valores de las cartas del jugador 2, n es la cantidad de cartas de cada jugador y k es la cantidad de cartas que debe sumar cada jugador. La función debe devolver 1 si gana el jugador 1, -1 si pierde el jugador 1 o 0 en caso de empate.

Dar y justificar la complejidad, no puede ser mayor a $O(n + k * \log(n))$

2. Dado un juego de 2 jugadores, donde cada jugador tiene N cartas y se juega a K rondas. En cada ronda los jugadores juegan la carta más grande, y gana quien tenga la mayor, luego los jugadores descartan la carta jugada. Al finalizar las K rondas, el ganador es el jugador que haya ganado más rondas. Se desea implementar una función:

```
int obtener_ganador(int* cartas_1, int* cartas_2, size_t n, size_t k);
```

Donde `cartas_1` representa un array con los valores de las cartas del jugador 1, `cartas_2` representa un array con los valores de las cartas del jugador 2, n es la cantidad de cartas de cada jugador y k es la cantidad de rondas. La función debe devolver 1 si gana el jugador 1, -1 si pierde el jugador 1 o 0 en caso de empate.

Dar y justificar la complejidad, no puede ser mayor a $O(n + k \log n)$

3. Se tiene una matriz de 1s y 0s donde un 1 representa una porción de tierra y un 0 una porción de agua. Se considera que dos porciones de tierra x e y pertenecen a una misma isla si es posible llegar desde x hasta y (también desde y hasta x) moviéndose entre porciones de tierras adyacentes (vertical, horizontal o diagonalmente). Se quiere saber la cantidad de islas.
 - a) Modelar el problema utilizando grafos. Explicar detalladamente cuáles serían los vértices y aristas.
 - b) Asumiendo que se cuenta con el grafo detallado en el punto b, implementar una función que reciba el grafo y resuelva el problema detallado anteriormente. Especificar y justificar la complejidad de la solución propuesta.
4. Se tiene una matriz en la cual un 1 representa una palta y un 0 un espacio vacío. Se sabe que si dos paltas están ubicadas juntas en la matriz (vertical u horizontalmente) y una de ellas está podrida, la otra eventualmente lo estará. Luego la nueva palta podrida, va a pudrir a sus vecinas y así sucesivamente. Se quiere saber la cantidad **mínima** de paltas que tiene que haber para que eventualmente esten todas podridas.
 - a) Modelar el problema utilizando grafos. Explicar detalladamente cuáles serían los vértices y aristas.
 - b) Asumiendo que se cuenta con el grafo detallado en el punto b, implementar una función que reciba el grafo y resuelva el problema detallado anteriormente. Especificar y justificar la complejidad de la solución propuesta.
5. En un monitor de dos colores (blanco y negro) la pantalla se puede entender como una matriz donde un 0 representa un bit en blanco y un 1 un bit en negro. Se entiende como una **figura**, a un grupo de bits en blanco próximos, esto es, que son vecinos horizontal o verticalmente. Para no sobrepasar a los usuarios de información, hay un máximo de figuras que un programador/a

puede poner en la pantalla simultáneamente, por lo que **se quiere tener una forma de obtener la cantidad de figuras blancas en pantalla en un momento dado.**

- a) Modelar el problema utilizando grafos. Explicar detalladamente cuáles serían los vértices y aristas.
- b) Asumiendo que se cuenta con el grafo detallado en el punto b, implementar una función que reciba el grafo y resuelva el problema detallado anteriormente. Especificar y justificar la complejidad de la solución propuesta.

6. Responde V/F justificando:

- a) Un grafo bipartito no puede tener ciclos.
- b) Si un grafo tiene algún orden topológico, al agregar un nodo con una arista que lo conecte a un nodo pre existente, el grafo sigue teniendo algún orden topológico.
- c) Para un grafo representado como matriz de incidencia, determinar el grado de entrada o de salida de un vértice es $O(V)$.

7. Responde V/F justificando:

- a) Si un grafo tiene un solo ciclo de longitud impar, se puede convertir en bipartito eliminando una de las aristas que componen el ciclo.
- b) Si existe el orden topológico en un grafo, entonces es único.
- c) Para un grafo representado como matriz de adyacencia, determinar el grado de entrada o de salida de un vértice es $O(V)$.

8. Responde V/F justificando:

- a) Si un grafo tiene más de un ciclo de longitud par, entonces no es bipartito.
- b) Si un grafo tiene nodos con grado de entrada 0, entonces existe el orden topológico.
- c) Para un grafo representado como lista de adyacencias (**lista de listas**), determinar el grado de entrada o de salida de un vértice es $O(V)$.