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

4.^{to} parcialito – 23/07/2021

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

Debés resolver 3 de los siguientes ejercicios. Uno entre el 1 y 3, otro entre el 4 y 6, otro entre el 7 y 9. Los ejercicios a resolver están definidos por tu padrón. Podés revisar en esta planilla los que te corresponden.

1. Se tiene un laberinto compuesto por casilleros. En dicho laberinto se permite el movimiento en cualquier dirección, pero moverse de un casillero a otro puede producir una cierta cantidad de cansancio. Cuánto cansancio produce un movimiento depende desde cuál casillero nos movemos, hacia cuál otro nos movemos (conocemos dichos valores).

   a. Explicar cómo se puede modelar este problema con un grafo.
   b. Diseñar un algoritmo (en pseudocódigo) que nos permita llegar desde un casillero inicial hasta el casillero final (salida, de la cual conocemos ya su posición), con el menor cansancio posible.
   c. ¿Cambiaría la elección del algoritmo si el cansancio fuera el mismo en cualquier dirección e igual para todos los casilleros? Si es así, explique por qué haría dicho cambio, y si no es así, explique por qué no resultaría conveniente.

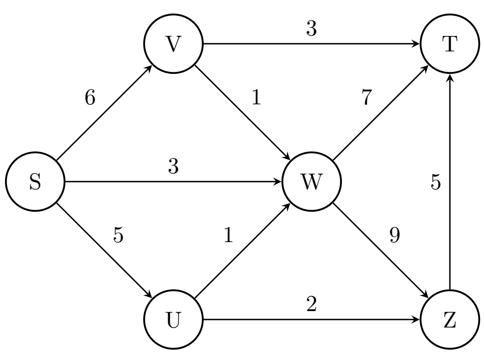
2. Grace viaja en auto desde un punto A hasta un punto B, pasando por calles, autopistas, colectoras o rutas. En diferentes puntos puede desviarse para agarrar una autopista, calle, colectora o ruta (según el caso, o hasta pudiendo tener más de una opción). Cada tramo que hace hasta llegar a la siguiente “posible bifurcación” le consume una cantidad de dinero (dependiendo del tramo). Esto puede ser por el pago de peajes, el combustible consumido en ese tramo y/o por haberse detenido en algún punto a comer las medialunas del famoso local de la ruta Algolaya. Grace cuenta con un dinero inicial de `$D`. Ella desea saber si existe una forma de llegar a destino (es decir, que no se detendrá por no poder pagar algún peaje, o no podrá cargar combustible) y, si es posible, minimizar la cantidad de dinero gastado al finalizar todo el trayecto.

   a. Explicar cómo se puede modelar este problema con un grafo.
   b. Diseñar (en pseudocódigo) un algoritmo que nos permita llegar desde el punto inicial hasta el destino de Grace, como es pedido.
   c. Supongamos que no existe forma posible para llegar hasta el destino con el dinero que cuenta ¿En qué punto de ejecución del algoritmo podemos ya darnos cuenta que no va a existir solución posible? Por ejemplo, si se olvidara la billetera en casa, Grace querría ejecutar el algoritmo y que inmediatamente le diga que no va a llegar a destino.

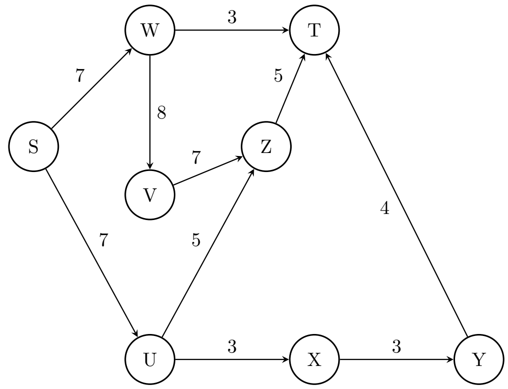
3. En un juego, un jugador comienza en un estado `"inicio"`, pasa por distintos estados (por ejemplo, `"entrenar para batalla"`, `"pasar a nivel 5"`, etc) y finalmente llega a un estado final llamado `"campeón del juego"`. Según la acción que realice un jugador puede pasar de un estado a otro. No se puede pasar de cualquier estado a cualquier otro. Algunos de esos pasajes le harán perder una cantidad de energía, y otros le harán ganar. No hay un máximo de energía que el jugador pueda tener (puede acumular para usar en futuras batallas).

   a. Explicar cómo se puede modelar este problema con un grafo.
   b. Diseñar un algoritmo (en pseudocódigo) que nos permita llegar desde el estado `"inicio"` hasta el estado `"campeón del juego"` con la mayor cantidad de energía posible.
   c. Explicar qué problemática podría suceder en este escenario, y cómo lo puede detectar el algoritmo planteado.

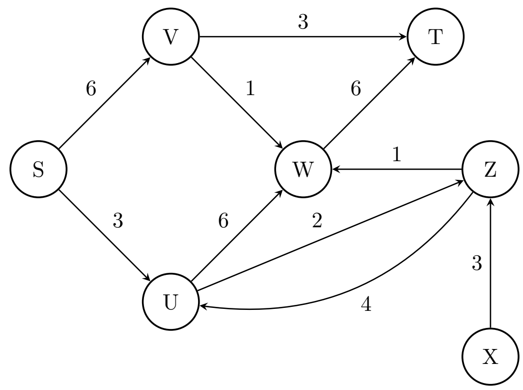
4. Realizar un seguimiento del Algoritmo de Ford-Fulkerson para obtener el flujo máximo de la siguiente red.



5. Realizar un seguimiento del Algoritmo de Ford-Fulkerson para obtener el flujo máximo de la siguiente red.



6. Realizar un seguimiento del Algoritmo de Ford-Fulkerson para obtener el flujo máximo de la siguiente red.



7. Definir si las siguientes afirmaciones son verdaderas o falsas. Justificar.

   a. Al resolver el problema de maximización de flujo buscamos caminos entre la fuente y el sumidero en la red residual. Si en dicho camino se utiliza una arista que no existe en el grafo original, significa que el flujo en la arista original (la del grafo original) se debe disminuir, por lo que el flujo total también disminuye.
   b. No podemos resolver el problema de maximización de flujo si hay ciclos en el grafo.
   c. Podemos calcular el Árbol de tendido mínimo de un grafo no dirigido que tenga pesos negativos.

8. Definir si las siguientes afirmaciones son verdaderas o falsas. Justificar.

   a. Al resolver el problema de maximización de flujo buscamos caminos entre la fuente y el sumidero en la red residual. Si en dicho camino se utiliza una arista que no existe en el grafo original, significa que no podemos considerar ese camino para aumentar el flujo total.
   b. Todo grafo dirigido con un vértice de grado de entrada 0, un vértice de salida 0 y una única componente débilmente conexa es Red de Flujo (sin necesidad de hacer ninguna otra modificación).
   c. Siempre podemos calcular el camino mínimo desde un vértice en un grafo dirigido que tenga pesos negativos.

9. Definir si las siguientes afirmaciones son verdaderas o falsas. Justificar.

   a. Al resolver el problema de maximización de flujo buscamos caminos entre la fuente y el sumidero en la red residual. Si en dicho camino se utiliza una arista que no existe en el grafo original, significa que el flujo en la arista original (la del grafo original) se debe aumentar.
   b. En un grafo no dirigido con pesos positivos, podemos aplicar el algoritmo de Bellman-Ford para obtener el camino mínimo desde un vértice hacia todos los demás.
   c. Al aplicar el algoritmo de Ford-Fulkerson para obtener el flujo máximo sobre una red de flujo, podemos aplicar cualquier tipo de método para obtener los caminos desde la fuente al sumidero, y no importa el tipo de recorrido, el camino encontrado siempre aumentará el flujo total.