Algorísmia QT 2014–2015

Examen final **12 de Enero de 2015**

Duración: 2h 30m

Instrucciones generales:

- El ejercicio 1 es obligatorio
- Debéis escoger dos de entre los ejercicios 2, 3 y 4. Debéis indicar claramente cuáles son los ejercicios elegidos. Si resolveis los tres ejercicios y la nota de los dos elegidos supera 4 puntos de los 6 posibles, se corregirá el tercer ejercicio y se sumará su nota a la nota de las otras preguntas contestadas.
- En cualquier caso, la calificación global de esta prueba escrita no podrá superar el 10, al sumarse todos los puntos Por ello, si la nota del apartado anterior supera a 10, la calificación global de la prueba escrita será 10.
- Para cada algoritmo que tengáis que desarrollar, garantizad que el algoritmo sea correcto (proporcione el tipo de solución requerida) y eficiente. Tenéis que proporcionar una descripción a alto nivel del algoritmo en pseudo-código, junto con las explicaciones y aclaraciones oportunas, de manera que se proporcionen: los argumentos que garantizan la corrección del algoritmo propuesto y el coste asintótico en caso peor del algoritmo (en función del tamaño de la entrada).
- Podéis hacer llamadas a algoritmos vistos en clase, pero si la solución es una variación de un algoritmo visto en clase tenéis que detallarlo.
- Se valorará especialmente la claridad del código (o pseudo-código) de las soluciones propuestas así como la de las explicaciones.
- Entregad por separado cada uno de los ejercicos y no olvidéis marcar en la hoja del problema 1 los dos problemas seleccionados (en el caso en que entregueis los tres).

- **Ejercicio 1 (4 puntos)**
  Responder a esta pregunta en la hoja en blanco con espacio para marcar las respuestas seleccionadas. Vuestra respuesta no puede utilizar más espacio y el resultado debe de ser legible.

  – Dado un grafo $G = (V, E)$ con pesos positivos en las aristas w_e, $e \in E$, y un conjunto $S \subseteq V$, sea $e = (u, v)$ una arista que tiene el peso mínimo entre S y $V - S$ y T un árbol de expansión de peso mínimo para G. La arista e forma parte de T? Justifica tu respuesta.

  – Explica las similitudes y diferencias entre los algoritmos de Bellman-Ford, Dijkstra y Floyd-Wharshall.

  – Explica qué es, qué propiedades tiene y para qué se utiliza, un paseo aleatorio en un grafo.

  – Explica las similitudes y diferencias entre un algoritmo tipo Monte-Carlo y uno tipo Las Vegas

- **Ejercicio 2 (3 puntos)** Tenemos n paquetes de video que se tienen que enviar secuencialmente por una única línea de comunicación (esto se denomina *streaming*). El paquete i-ésimo está formado por b_i bits y tarda en ser enviado t_i segundos. Tanto b_i como t_i son enteros positivos. Como consecuencia del modelo de streaming no se pueden enviar dos paquetes a la vez. Tenemos que decidir una planificación, es decir el orden en que enviar los paquetes, asumiendo que no hay retardo entre el final del envio de un paquete y el principio del envio del siguiente. Asumiremos que la transmisión se inicia en el instante 0. Observad que con estas condiciones, para cualquier planificación, se inicia la transmisión a tiempo 0 y esta termina a tiempo $\sum_{i=1}^{n} t_i$.

  Además el proveedor de la conexión quiere usar un ancho de banda que no sea demasiado grande. Esta situación nos lleva a considerar un parámetro adicional r y la restriccion que, para cada natural t, el número total de bits que se envian en el intervalo [0,t] no puede superar rt. Observad que esta restricción no dice nada sobre intervalos de tiempo que no empiezan en 0 ni sobre intervalos que empiezan en 0 y termian en un valor no entero.

Diremos que una planificación es válida si satisface la restricción previa. El problema que queremos resolver els el siguiente: dado un conjunto de n paquetes, cada uno representado por el par (b_i, t_i) y el valor del parámetro r, determinar si existe una planificación válida.

Por ejemplo, si $n = 3$, con $(b_1, t_1) = (2000, 1)$, $(b_2, t_2) = (6000, 2)$ y $(b_3, t_3) = (2000, 1)$, y $r = 4000$, la planificación 1, 2,3 es válida. A tiempo $t = 1$ se han enviado 2000 bits, a tiempo $t = 2$ un total de 5000 bits (2000 del primer paquete y la mitad del segundo), a tiempo $t = 3$ son 8000 bits y a tiempo $t = 4$ 10000.

Para resolver este problema tenéis que resolver los apartados siguientes:

1. Determinar si el enunciado:

   existe una planificación válida, si y solo si, para cada paquete i, se cumple $b_i \leq r\, t_i$

   es o no correcto. Proporcionad una demostración en caso de que lo sea o un contraejemplo si no lo es.

2. Proporcionad un algoritmo que, para una entrada de n paquetes, especificados por (b_i, t_i) y el valor del parámetre r, determine si existe una planificación válida. Vuestro algoritmo debería tener complejidad polinómica en n.

- **Ejercicio 3 (3 puntos)**

  Un grupo de miembros de la asociació astronómica de la UPC quiere observar el cielo el próximo sábado ya que se espera una noche completamente despejada y sin luna. Supongamos lo siguiente:

  – Hay n eventos que ocurren durante una intervalo de n minutos, el suceso j ocurre en el minuto j. No hay dos sucesos que ocurran al mismo tiempo. El evento j sólo se puede observar en el minuto en que ocurre.

  – Utilizamos un sistema 1-dimensional como modelo del cielo, utilizando la coordenada que corresponde con el grado del ángulo del teléscopio que nos permite visualizar esta posición. En consecuencia, el suceso

j tiene asociada una coordenada d_j, podeis asumir que debido a la precisión d_j es un valor entero.

- En el minuto 0, la posición del telescopio es la 0.
- El último evento es más importante que el resto. Por ello el grupo requiere que sea observado.

El teléscopio del departamento es muy grande y solo se puede mover un grado cada minuto. Por tanto es posible que no se puedan observar todos los eventos del intervalo de tiempo seleccionado. Sea S un conjunto de sucesos. Los eventos $j, k \in S$ son *consecutivos*, si $j < k$ y no hay eventos observables en S entre los minutos j y k, S es *observable*, si para a cada par de eventos consecutivos $j, k \in S$ se cumple que el telescopio tiene el tiempo necesario para moverse (con velocidad como máximo 1 grado por minuto) de d_j a d_k de manera que se puedan observar los dos sucesos.

Dados los n sucesos mediante una secuencia $(d_j)_{1 \leq j \leq n}$ de coordenadas, queremos encontrar el conjunto S de eventos observables que incluya el evento n y con el número más alto posible de eventos.

Ejemplo: Si tenemos $n = 9$ y la secuencia de coordenadas

$$(1, -4, -1, 4, 5, -4, 6, 7, -2),$$

la solución óptima es $S = \{1, 3, 6, 9\}$. Observad que sin la restricción de que el evento 9 tenga que ser observado el conjunto $\{1, 4, 5, 7, 8\}$ es el conjunto observable con mayor número de sucesos.

Dada una secuencia de n enteros describiendo las coordenadas de los n eventos proporcionad un algoritmo, con coste polinómico, que permita obtener el conjunto observable más grande S con la condición $n \in S$.

Tened en cuenta que debido a la velocidad del telescopio, la posicion indicada por el mismo, en cualquier instante de tiempo, podeis suponer que se encuentra dentro del intervalo $[-n, n]$.

- **Ejercicio 4 (3 puntos)**
  El *problema de la evacuación* tiene la siguiente definición. Nos dan un grafo dirigido $G = (V, E)$ que representa una red de carreteras. Una cierta colección de nodos $X \subseteq V$ se designan como *nodos habitados*. Otra colección de nodos S son designados como *nodos seguros*. Suponemos que S y X son disjuntos. En el caso de una emergencia, queremos conocer rutas de evacuación desde los nodos habitados a los seguros. Un conjunto de *rutas de evacuación* se define como un conjunto de caminos en G tales que: (i) cada nodo en X aparece como el inicio de un camino, (ii) el último nodo en un camino aparece en S, (iii) los caminos no comparten aristas.

  Cada ruta de evacuación proporciona una ruta a travéss de la cual los habitantes de un nodo poblado pueden escapar a un nodo seguro sin compartir ninguna parte del camino con habitantes de otros nodos.

  Un conjunto de *rutas de evacuacin mixtas* se define como un conjunto de caminos en el que (i) cada nodo en X aparece como el inicio de un camino, (ii) el último nodo en un camino aparece en S.

  Este tipo de rutas proporciona rutas de escape en las que parte del camino es compartido entre habitantes de distintas ciudades.

  Supongamos que además, debido a las restricciones de tráfico, cada arco $e \in E$ tiene asignada una capacidad de tráfico $c(e)$ que corresponde al número máximo de vehículos que pueden atravesarlo. Una petición de tráfico $r(x)$, para un nodo habitado x, representa el número de vehículos que se quieren evacuar desde x.

  - Dados $G = (V, E, c)$ y $S, X \subseteq V$, muestra como decidir en tiempo polinómico si un conjunto de rutas de evacuación existe o no. En caso de que si que existan, el algoritmo debe proporcionar, para cada nodo poblado, la capacidad del arco de capacidad mínima de la ruta que se inicia en él.
  - Dados $G = (V, E, c)$, $S, X \subseteq V$, y una petición de tráfico $r(x)$, para cada $x \in X$, muestra como decidir en tiempo polinómico si existe un conjunto de rutas mixtas de evacuación que permitan enviar los vehículos solicitados desde X a nodos seguros y que, además, para cada arco, se cumpla la condición de que el número de vehículos que lo atraviesan no supere la capacidad de tráfico del arco. En caso de que sea posible la evacuación con rutas mixtas, el algoritmo tiene que

devolver además, para cada nodo $x \in X$, un plan de evacuación indicando: para cada vehículo sliendo de x, un camino con origen en x y final en un nodo seguro. El total de tráfico asignado entre todos los caminos del plan de escape de todos los nodos $x \in X$ tiene que cubrir la totalidad de las peticiónes de tráfico y cumplir las restricciones de capacidad de los arcos.

Apellidos	Nombre	DNI

Problemas seleccionados | | |

Ejercicio 1